

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

AMAZON.COM, INC., AMAZON.COM SERVICES LLC,

Petitioner,

v.

NOKIA TECHNOLOGIES OY,

Patent Owner.

Case No. IPR2024-01507

U.S. Patent 8,996,693

Declaration of Mark E. Crovella

TABLE OF CONTENTS

	<u>Page</u>
I. Overview of the Technology	4
A. Static and Dynamic Processing	5
B. Batch Processing	7
C. Marshalling and Unmarshalling Data	8
II. The '693 Patent.....	9
A. Overview	9
B. Prosecution History	12
C. Priority Date	14
D. Challenged Claims	14
III. Level of Ordinary Skill in the Art	15
IV. Claim Construction.....	18
A. “processing element”.....	18
B. “dynamic processing mechanism”	19
C. “static processing mechanism”.....	20
V. Invalidity Grounds	21
A. Grounds 1 and 2: Obviousness over Foster Alone (Ground 1) and Foster Combined with Williams (Ground 2).....	21
1. Foster et al., <i>IBM InfoSphere Streams: Assembling Continuous Insight in the Information Revolution</i> (2011) (Ex-1004, “Foster”).....	22
2. Williams et al., <i>Implementing CICS Web Services</i> (2007) (Ex-1005, “Williams”).....	25
3. Independent Claim 1	28
4. Independent Claim 11	55

5.	Dependent Claims 2 and 12	61
6.	Dependent Claims 3 and 13	69
7.	Dependent Claims 4 and 14	71
8.	Dependent Claims 5 and 15	74
9.	Dependent Claims 7 and 17	77
10.	Dependent Claims 8 and 18	78
11.	Dependent Claims 9 and 19	81
12.	Dependent Claims 10 and 20	85
VI.	Educational Background and Qualifications	88
A.	Compensation.....	91
B.	Materials and Other Information Considered	91
VII.	Understanding of the Law	91
VIII.	Reservation of Rights	91
	Appendix 1: Curriculum Vitae of Mark E. Crovella.....	93
	Appendix 2: Materials Considered in the Preparation of this Declaration	116
	Appendix 3: Challenged Claims.....	117
	Appendix 4: Understanding of the Law	123

I, Mark E. Crovella declare as follows:

1. I am a Professor in the Department of Computer Science, a member of the Faculty of Computing & Data Sciences, where I am Chair of Academic Affairs, at Boston University. I have prepared this report as an expert witness retained by Amazon.com, Inc. In this report I give my opinions as to whether certain claims of U.S. Patent No. 8,996,693 (“the ’693 patent”) are invalid. I provide technical bases for these opinions as appropriate.

2. This report contains statements of my opinions formed to date and the bases and reasons for those opinions. I may offer additional opinions based on further review of materials in this case, including opinions and/or testimony of other expert witnesses. I make this declaration based upon my own personal knowledge and, if called upon to testify, would testify competently to the matters contained herein.

I. OVERVIEW OF THE TECHNOLOGY

3. The technology at issue relates to data processing. Dynamic data processing refers to processing data that is in motion, such as detecting contextual trends in an incoming live stream of Tweets and social media posts. Static data processing refers to processing data that is at rest, such as analyzing a collection of historic sales and demographic data to predict demand for a product in a new market. Even before the ’693 patent was filed in 2012, persons of skill in the art

were combining both approaches to provide better data analysis. For example, in the automated stock trading domain, static processing was used to generate models that were applied to the dynamic processing of incoming stock market events to determine whether to carry out trades; these models were updated periodically with newly accumulated data to improve their performance. Ex-1011, Abstract.

A. Static and Dynamic Processing

4. By 2012, various mechanisms were known and used for static processing. Static processing hails back to at least the 1960s when “hierarchical databases” were invented to “capture[] and store[] that data” for post facto processing “to captur[e] and understand[] what has happened.” Ex-1004, §1.1.3 at 00034. These “online transaction processing (OLTP) systems” laid the path for “online analytic processing (OLAP)” that used techniques such as data mining on the stored data to “understand[] why things happened to make more informed recommendations.” Ex-1004, §1.1.3 at 00035. The commonality of these traditional approaches is that “the data is stored,” and therefore the processing occurs on static, accumulated data. *Id.*

5. Advances in processing power, communication bandwidth, storage, and the proliferation of information sources increased the amount and timeliness of data available for processing. Ex-1004, §1.1 at 00022-24, §1.1.2 at 00029-32. Such real-time data was “considered streaming data or data in motion.” Ex-1004,

§1.1 at 00025. The flood of real-time information and increased processing power and techniques gave birth to dynamic processing in order to provide “faster analytics than is possible when you have to wait until the data is retrieved from storage.” Ex-1004, §1.1.3 at 00036. Mechanisms and techniques for this type of dynamic processing were often grouped under the moniker of “real-time analytic processing (RTAP).” Ex-1004, §1.1.3 at 00036.

6. Thus, two computing paradigms were divided based on the temporal nature of the data: the traditional static processing of “data-at-rest (historical data)” and the newer dynamic processing of “data-in-motion (current data).” Ex-1004, §1.1.4 at 00037, §1.1 at 00025-26, Fig. 1-2 at 00025:



Figure 1-2 Comparing traditional and stream computing

Because the data processed was either real-time data or accumulated slow-moving data, these processing techniques could also be called “real-time” or “slow-moving” processing, respectively.

7. Further development in the data processing field led to the use of both of these processing methods together. *E.g.*, Ex-1010, Abstract (“A mechanism for using stream objects in stream processing...”), ¶33 (processing “real-time data,

pseudo real-time data and/or archived data.”). Various systems and methods were employed to combine the results of static and dynamic processing to provide the “seamless processing of current and historical data.” Ex-1004, §7.1 at 00338; *see, e.g.*, Ex-1004, Fig. 7-1 (“real-time and historical insights”):

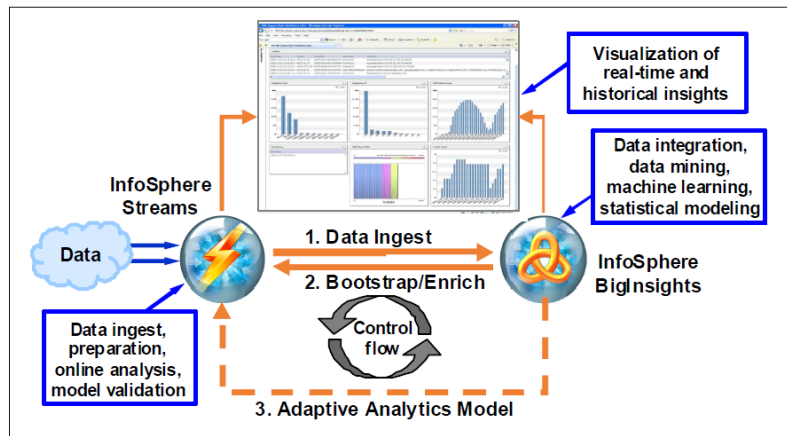


Figure 7-1 Big Data application scenarios

See also, e.g., Ex-1011, §1 (“[T]he data profile, from being composed entirely of the data at rest (stored data) or the data in motion (streaming data) is now some combination of the two. Examples include, finance applications like automated stock trading [8], scientific applications like the square kilometer array [20], network security and monitoring applications like the ones used for intrusion detection [17]”) (brackets in original).

B. Batch Processing

8. The claims of the ’693 patent refer to a batch mode of static processing. Batch processing was a common technique used for static processing, because slow-moving data would accumulate over time. Such data would be

processed in batches, i.e., a number of data items grouped together for intermittent processing. Batch processing was used in various algorithms and methods. *E.g.*, Ex-1014, ¶133. Batch processing could be scheduled by frequency or periodicity. For example, “[t]he processing of batches can be daily, weekly, or monthly, depending on the volume of transactions and other considerations.” Ex-1013, 00019, 00015. Map-reduce was a well-known static processing algorithm commonly run on periodic batches. Ex-1012, §1 at 00001 (“In MapReduce environments, many production jobs are run periodically on new data.”), §8 at 00009 (“MapReduce (and its open source implementation Hadoop) was designed for periodically running large batch workloads.”); Ex-1015, ¶6 (“MapReduce was originally aimed at large, generally periodic, production batch jobs....”).

C. Marshalling and Unmarshalling Data

9. The claims of the ’693 patent refer to marshalling and unmarshalling data. Marshalling and unmarshalling data into and from formats suitable for transmission was well-known. Marshalling was a generic term for gathering data from one process and converting it into a format that can be used either for storage or for transmission to another process. Ex-1005, §1.4.3 at 00039; *see also, e.g.*, Ex-1015, 000064 (“Marshalling involves converting local objects into portable form so that they can be transmitted to a remote process.”). It was well-known, as the term suggests, that unmarshalling was the inverse process to undo the

conversion and extract the original data. *See* Ex-1016, ¶2 (“a process (marshalling) for converting from a format which is specific to the architecture of particular computer, particular operating system, and particular program language into a neutral data stream and *a reverse process (unmarshalling)*”).

10. JSON (JavaScript Object Notation) was a common format into which data was marshalled. It is a standard, text-based format for describing data based on the JavaScript object syntax that was first officially described in RFC4627. Ex-1017, 00001. Because it uses conventions familiar to users of C-like programming languages such as C, Java, Perl, Python, etc., it was commonly used as a data-interchange format. XML was another common format for marshalling data for online applications. Ex-1013, 00025.

II. THE '693 PATENT

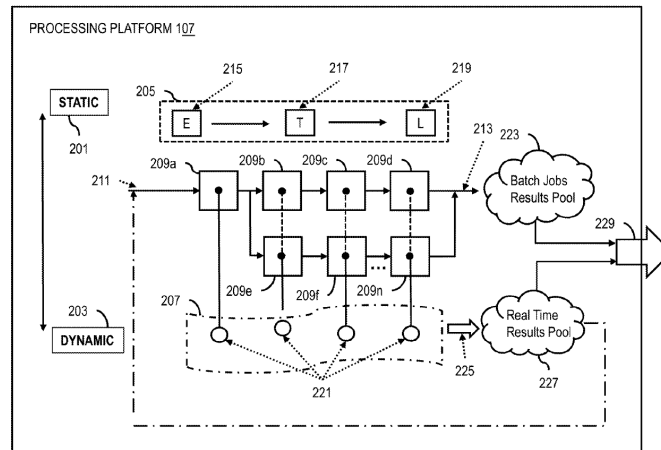
A. Overview

11. The '693 patent provides a “method and apparatus for providing dynamic stream processing of data based on static analytics.” Ex-1001, (54) Title. The alleged invention is directed to “a method, apparatus, and computer program for smart data processing methods.” Ex-1001, 3:35-47. The '693 patent does not purport to invent “dynamic processing mechanisms,” “static processing mechanisms,” or “processing elements.” Ex-1001, Abstract. Rather, it acknowledges “data processing architectures and techniques that traditionally have

been used” for the processing of both “‘slow moving’ data (e.g., static or relatively static data)” and “data [] collected as data streams in real-time.” Ex-1001, 1:22-36. The ’693 patent also notes that “[t]raditionally, data processing and analysis may be implemented via one or more servers (or nodes) and/or clusters of servers (or nodes) that provide, for instance, distributed computing and/or data storage to support the services.” Ex-1001, 1:18-21.

12. The ’693 patent discusses the “need for data processing methods to generate comprehensive results that combines the features of a slow moving data systems with real-time data systems.” Ex-1001, 1:40-42. The patent purportedly solves the “technical challenges to managing and/or integrating the data processing and analysis methods for slow moving data with the data processing and analysis methods for real-time data.” Ex-1001, 1:32-36. Specifically, the patent proposes “introduc[ing] the capability for utilization of static and dynamic data analytics in various data processing mechanisms.” Ex-1001, 4:23-49. Figure 2 illustrates a proposed arrangement to combine the two approaches to processing data. Ex-1001, 11:6-12:60.

FIG. 2



Ex-1001, Fig. 2. According to the patent, this provides the benefit of generating “more accurate and comprehensive results.” Ex-1001, Abstract, 3:35-52.

13. However, as explained above, processing and analyzing slow-moving data and real-time data was already known in the art, as well as techniques for combining the two processing methods. *Supra* §I.A. The ’693 patent merely recites mundane steps using well-known techniques in the field of data processing and analysis, that amount to little more than determining data to process, preparing the data for transfer to or use at a different system, transferring the data, processing the data, and transferring it back. *See, e.g.*, Ex-1001, 25:31-44 (claim 1), Abstract, 1:43-24. In short, the ’693 patent “combines the features of a slow moving data systems with real-time data systems” by sending data back and forth to be processed by both types of systems. Ex-1001, 1:40-42.

14. Indeed, the single point of novelty the Applicant argued during prosecution was that the prior art transferred the data processed at the static

processing application to another dynamic processing application instead of the original dynamic processing application that it came from. *Infra* §II.B. As I explain below in detail, transferring the data back to the original dynamic processing mechanism was known in the art. *Infra* §VI.A.3.[1e].

B. Prosecution History

15. The '693 patent issued March 31, 2015 from Application No. 13/621,511 ("the '511 application"). I understand that the '693 patent contains no priority claim to a prior filing and that there are no other patents or applications in the '693 patent's family. It was allowed after a single non-final office action rejecting the claims under 35 U.S.C. §102, and a §112 objection to the preamble.

16. In the only office action, the Examiner rejected claims 1-20 as anticipated by the prior art reference Siripurapu. Ex-1002, 00124-127. As-filed independent claim 1 only recited the determining and marshalling limitations of the issued patent and specified that the marshalled data object is processable, instead of claiming processing, by a static processing mechanism:

1. A method comprising facilitating a processing of and/or processing (1) data and/or (2) information and/or (3) at least one signal, the (1) data and/or (2) information and/or (3) at least one signal based, at least in part, on the following:

at least one determination of at least one processing element of at least one dynamic processing mechanism; and

a marshalling of the at least one processing element as at least one data object,

wherein the at least one data object is processable by at least one static processing mechanism.

Id., 00039.

17. To overcome the rejection, claim 1 was amended to further recite transferring the data object to a static processing mechanism, processing the data object by the static processing mechanism, and transferring the processed data object back to the dynamic processing mechanism:

1. (Currently Amended) A method comprising: ~~facilitating a processing of and/or processing (1) data and/or (2) information and/or (3) at least one signal, the (1) data and/or (2) information and/or (3) at least one signal based, at least in part, on the following:~~

~~at least one determination of at least one processing element of at least one dynamic processing mechanism; and~~

~~a marshalling of the at least one processing element as at least one data object,~~

~~wherein the at least one data object is processable by at least one static processing mechanism.~~

determining via a processor at least one processing element of at least one dynamic processing mechanism;

causing, at least in part, a marshalling via the processor of the at least one processing element

as at least one data object:

transferring of the at least one data object to at least one static processing mechanism;

processing of the at least one data object by the at least one static processing mechanism; and

after processing, transferring the at least one data object back to the at least one dynamic processing mechanism.

Ex-1002, 00136 (text formatting as in original).

18. Applicant argued that Siripurapu transfers the processed data to a *different* streaming application, rather than “back to” the *original* application that marshalled and transferred the data in the first place. Ex-1002, 00147-148.

19. A notice of allowance issued without specifying a reason for allowance. Ex-1002, 00169-182.

C. Priority Date

20. The '511 application that issued as the '693 patent was filed on September 17, 2012, claiming priority to itself. I have been asked to assume that the '693 patent is entitled to its claimed priority date of September 17, 2012.

D. Challenged Claims

21. I understand that Petitioners are challenging the validity of claims 1-5, 7-15, and 17-20 of the '693 patent in the Petition for Inter Partes Review to which this declaration will be attached. Those claims are reproduced in Appendix 3.

While this Petition and declaration are directed to the challenged claims, I have considered all 20 claims of the '693 patent, as well as portions of the '693 patent prosecution history in forming my opinions.

III. LEVEL OF ORDINARY SKILL IN THE ART

22. I have analyzed the '693 patent and determined that the field of the patent is “data processing and analysis.” The specification explains that “[o]ne area of interest has been development of applications and services, which *process and/or utilize various types of data*¹ in order to provide accurate, appropriate, and updated information.” Ex-1001, 1:6-36. And that “these applications and services can include *processing and analyzing data* utilized, for example, by social networking services, navigation services, search engines, content providers, and the like.” *Id.* As I explained above, the '693 patent attempts to address the “need for data processing methods to generate comprehensive results that combines the features of a slow moving data systems with real-time data systems” in this field. Ex-1001, 1:40-42; *supra* §II.A. The '693 patent’s alleged invention itself is directed at “smart data processing methods.” Ex-1001, 3:35-47, Ex-1001, 3:6-33 (describing figures of embodiments as “smart data processing platform,” “smart data processing architecture,” “processing platform,” etc.).

¹ All emphasis is added unless otherwise noted.

23. In determining the characteristics of a hypothetical person of ordinary skill in the art (“POSITA”) of the ’693 patent at the time of the claimed invention, I considered several things, including various prior art techniques relating to the data processing and analysis, the type of problems that such techniques gave rise to, and the rapidity with which innovations were made. I also considered the sophistication of the technologies involved, and the educational background and experience of those actively working in the field at the time. I also considered the level of education that would be necessary to understand the ’693 patent. Finally, I placed myself back in the relevant period of time and considered the engineers and programmers that I have worked with and managed in the field of data processing and analysis.

24. I came to the conclusion that a person of ordinary skill in the field of art of the ’693 patent would have been a person with a bachelor’s degree in computer science, electrical engineering, computer engineering, or a similar field with at least two years of experience in data processing and analysis or a person with a master’s degree in mechanical engineering, electrical engineering, computer science, or a similar field with a specialization in data processing and analysis. A person with less education but more relevant practical experience may also meet this standard.

25. In the 2012 timeframe, a POSITA would have known and had the skills necessary to:

- Design/implement data processing systems, including distributed computing systems;
- Implement techniques for static data analysis and real-time data analysis (including, e.g., known techniques such as MapReduce, sentiment analysis, social network analysis, association rule learning, clustering, machine learning, stream processing);
- Design and use data structures and models (including those for Big Data analysis and distributed computing);
- Use database warehousing, OLTP (On-Line Transactional Processing), OLAP (On-Line Analytical Processing), RTAP (Real-time Analytic Processing) tools and protocols;
- Marshall and unmarshall (e.g., serialize and deserialize) data;
- Implement known algorithms to process data; and
- Design/implement displays and interfaces.

IV. CLAIM CONSTRUCTION

A. “processing element”

26. Based on my review of the claims and specification of the '693 patent, it is my opinion that a POSITA would have understood that “processing element” means “data to be processed.”

27. The '693 patent does not limit its discussion or scope of processing elements to another, well-known use of the term as an element that carries out processing (e.g., a processing mechanism). Instead, independent claim 1 uses processing element (PE) in the reverse sense as data to be processed. Ex-1001, 25:33-44 (“determining...at least one processing element...; marshalling...of the at least one processing element as at least one data object; processing of the at least one data object”). The specification explains that PEs are destined to be processed. Ex-1001, 12:32-48 (“[T]he data is loaded into an end target (e.g., data warehouse, service provider, intermediate application, etc.).”).

28. The '693 patent does not limit a PE to any specific type of data; rather, a PE “may include computational *code* and/or *data*, which may be associated with one or more information items from a user and/or a service provider.” Ex-1001, 6:63-7:1, 7:1-16 (“[A] PE may include data....”). Examples include “data feeds, users, service providers (e.g., Twitter®), search engines, content providers, etc.” *Id.*, 3:67-4:5, 6:58-7:1 (same), 15:54-61 (same).

B. “dynamic processing mechanism”

29. Based on my review of the claims and specification of the ’693 patent, it is my opinion that a POSITA would have understood that “dynamic processing mechanism” means a “mechanism for processing data in real-time or near real-time.”

30. The ’693 patent explains that “data processing mechanisms (e.g., engines, pipelines, etc.) may include a dynamic stream processing (e.g., *real-time or a near real-time*)... where the dynamic stream process may operate on one or more data feeds from various sources (e.g., users, service providers, sensors, etc.).” Ex-1001, 5:4-11. And dynamic processing carried out by such mechanisms is considered a synonym to “stream processing.” Ex-1001, 3:67-4:5 (“*[I]n the dynamic processing (e.g., stream processing), the data may be processed near real-time, wherein the data may be considered to be fast-moving (e.g., near real-time) and from various sources, for example, data feeds, users, service providers (e.g., Twitter®), search engines, content providers, etc.*”).

31. The ’693 patent’s embodiments confirm this understanding, explaining that “the dynamic processing mechanism 601 receives *real-time and/or near real-time data* from one or more data feeds 605 (e.g., one or more data and/or service providers), where the data *is processed by a stream processing engine 607* (e.g., stream processing framework, context mapping algorithms and data-

structures, etc.).” Ex-1001, 16:43-56, 17:9-17 (“[T]he dynamic processing mechanism 601 operates in substantially real-time....”), Figs. 6-7 (showing stream processing engine 607 in dynamic processing mechanism 601 receiving and processing real-time data feed 605).

C. “static processing mechanism”

32. Based on my review of the claims and specification of the ’693 patent, it is my opinion that “static processing mechanism” means a “mechanism for processing for processing historical or slow-moving data.”

33. The ’693 patent explains that “data processing mechanisms (e.g., engines, pipelines, etc.) may include... a static processing (e.g., map-reduce batch jobs)...where...the static process may operate on data from one or more databases (e.g., historical data).” Ex-1001, 5:4-11. “[H]istorical data may include large volumes of contextual data collected from various sources over time (e.g., hours, days, weeks, etc.).” Ex-1001, 5:11-17. Because static processing analyzes “slow-moving data,” they operate, for example, periodically in large batches. Ex-1001, 3:58-67 (“In the static processing method, as the database sizes may reach multiple gigabytes and terabytes, the data is processed according to one or more predetermined schedules (e.g., daily/weekly batch jobs), wherein the data may be considered to be slow-moving data and available from various sources (e.g., data stores.)”).

34. Dependent claims 9/19 and 10/20 recite limitations consistent with this understanding, which is further confirmed by the description of various embodiments in the specification. Ex-1001, 26:25-37 (claiming “batch mode with a predetermined batch frequency” (claim 9) and “performing...slow moving data analytics” (claim 10)), 28:13-25 (same for claims 19/20), 7:24-35 (same), 7:61-8:2 (static processing mechanisms (e.g., batch jobs)), 16:45-48 (same), 17:12-24, 16:56-6 (“a map reduce engine 609 to manage and process data”), Figs. 6-7 (“Map Reduce Engine 609”).

V. INVALIDITY GROUNDS

35. Based on my review and analysis of the materials cited herein, my opinions regarding the understanding of a POSTA in the 2012 timeframe, and my training and experience, it is my opinion that the challenged claims of the '693 patent are invalid based on the following grounds:

Grounds	Claims	Statutory Basis	Prior Art
1	1-5, 7-15, 17-20	§103	Foster
2	1-5, 7-15, 17-20	§103	Foster in view of Williams

The reasons for my conclusions are explained more fully below.

A. Grounds 1 and 2: Obviousness over Foster Alone (Ground 1) and Foster Combined with Williams (Ground 2)

36. Foster alone, and combined with Williams teaches all limitations of the challenged claims.

1. Foster et al., *IBM InfoSphere Streams: Assembling Continuous Insight in the Information Revolution* (2011) (Ex-1004, “Foster”)

37. I have reviewed the Foster reference. I understand that Foster was not cited or considered during prosecution of the '693 patent, based primarily on the fact that Foster is not cited on the face of the patent, nor have I seen the reference discussed in the prosecution history.

38. Foster is a publication in the IBM® Redbooks® series entitled “IBM InfoSphere Streams: Assembling Continuous Insight in the Information Revolution” that was published in October 2011. Ex-1008, ¶4; Ex-1006, 000009. I understand that Foster is prior art under pre-AIA §102(a) because it was published before the invention of the '693 patent, which claims priority to an application filed September 17, 2012.

39. Declarations from co-author Michael Spicer and from the Internet Archive confirm that Foster is a printed prior art publication. Ex-1008; Ex-1006. The publication date of October 2011 found in the front matter of Foster (Ex-1004, 00003-4) is corroborated by the coauthor and by IBM's website, as archived by the Internet Archive. Ex-1008, ¶4; Ex-1006, 000009 (summary page for Foster, listing publish date of “10 October 2011” and including a link to download Foster). The Internet Archive itself found and archived a copy of Foster no later than June 3,

2012 (Ex-1006, 00011-467), which is before the earliest filing date of the '693 patent.

40. IBM Redbooks, including Foster, were publicly accessible. IBM Redbooks is a series covering technical information on computing products, platforms, and solutions that was authored by researchers and engineers at IBM, and intended to be consumed by engineers in the fields of computing and data processing. Ex-1008, ¶¶5-6. IBM Redbooks were well-known among and consulted by engineers in the fields of computing and data processing systems. *Id.*, ¶7. They were distributed to the public in electronic and hard copies, and were available for download at no cost on IBM's website. *Id.*, ¶6. According to IBM's website, as archived by the Internet Archive on November 1, 2011, IBM Redbooks were highly popular, with over "500,000 downloads every month" and "80,000+ technical professionals subscribe[d] to the IBM Redbooks Weekly Newsletter" at the time. Ex-1006, 00005. The IBM Redbooks website was, and still is, searchable by keyword. *See* Ex-1006, 00007.

41. Foster is directed to Big Data analysis techniques and tools running on distributed computing systems. Ex-1004, §1.1.4 at 00037. In particular, Foster discusses the implementation and operation of its "IBM InfoSphere Streams and IBM InfoSphere BigInsights" products. Ex-1004, §1.1.4 at 00037. Streams "is a product architected specifically to help clients continuously analyze massive

volumes of streaming data at extreme speeds.” Ex-1004, §1.2 at 00037-51.

Streams was paired with BigInsights, which provides “data-at-rest (historical) analysis.” Ex-1004, §1.1.4 at 00037. BigInsights integrated industry standard static data processing technology with IBM’s own homegrown technology. Ex-1004, §1.1.4 (“BigInsights delivers an enterprise-ready big data solution by combining Apache Hadoop, including the MapReduce framework and the Hadoop Distributed File Systems (HDFS), with unique technologies and capabilities from across IBM.”).

42. Foster describes, among other potential applications, “the approach of integrating Streams with...BigInsights.” Ex-1004, 00337-345:

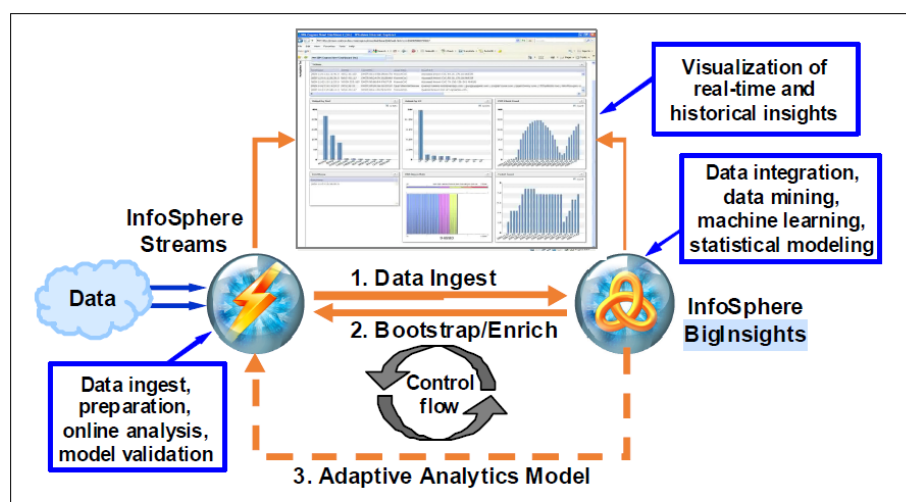


Figure 7-1 Big Data application scenarios

Fig. 7-1 at 00339. In particular, Foster explains how incoming real-time streams of data can be prepared at Streams for ingestion to BigInsights, and further processed at BigInsights to enrich the data and to build adaptive analytics models

that are then used for the ingestion and analytics at Streams. Ex-1004, §7.1.1-4 at 00338-341. Foster explains the “interactions between Streams and BigInsights” to provide these functions, using a number of examples. Ex-1004, §7.1.6 at 00343, §7.1.5-6 at 00342-344 (discussing use case “tracking positive and negative sentiment being expressed about products”), §7.1.2-4 at 00339-341 (discussing uses cases of “Call Detail Record (CDR) processing” and “social media based lead-generation”).

43. Foster is analogous art because it is in the same field of endeavor as the '693 patent, which is data processing and analysis. *Supra* §III. Foster discusses technologies related to data analytics, including “real-time analytic processing (RTAP).” Ex-1004, §I at 00021, §1.1.3 at 00034-36. Foster presents examples of solutions that combine real-time analysis with static analysis. Ex-1004, §1.2.3 at 00047-51. Foster is reasonably pertinent to the problem addressed by the '693 patent's inventor, because it attempts to solve the similar need for “products and tools designed to” “analyze and take action in real time” using an ever-growing volume of real-time data Ex-1004, §1.1.1-2 at 00026-33.

2. Williams et al., *Implementing CICS Web Services* (2007) (Ex-1005, “Williams”)

44. I have reviewed the Williams reference. I understand that Williams was not cited or considered during prosecution of the '693 patent, based primarily

on the fact that Williams is not cited on the face of the patent, nor have I seen the reference discussed in the prosecution history.

45. Williams is a publication in the IBM[®] Redbooks[®] series entitled “Implementing CICS Web Services” that was published in October 2007. Ex-1009, ¶4; Ex-1006, 000005. I understand that Williams is prior art under pre-AIA §102(a) because it was published before the invention of the ’693 patent (which claims priority to an application filed September 17, 2012) and under §102(b) because it was published more than one year before the earliest filing date of the ’693 patent.

46. Declarations from co-author Michael Ebbers and from the Internet Archive confirm that Foster is a printed prior art publication. Ex-1009; Ex-1007. The publication date of December 2006 found in the front matter of Williams (Ex-1005, 00003-4) is corroborated by the coauthor and by IBM’s website, as archived by the Internet Archive. Ex-1009, ¶4; Ex-1007, 000005 (summary page for Williams, listing publish date of “12 December 2006” and including a link to download Foster). The Internet Archive itself found and archived a copy of Foster no later than February 21, 2007 (Ex-1007, 00007-601), which is more than one year before the earliest filing date of the ’693 patent.

47. IBM Redbooks, including Williams, were publicly accessible. IBM Redbooks is a series covering technical information on computing products,

platforms, and solutions that was authored by researchers and engineers at IBM, and intended to be consumed by engineers in the fields of computing and data processing. Ex-1009, ¶5. IBM Redbooks were well known among and consulted by engineers in the fields of computing and data processing systems. *Id.*, ¶6. They were distributed to the public in electronic and hard copies, and were available for download at no cost on IBM’s website. *Id.*, ¶5. As I explained above for Foster, according to IBM’s website, as archived by the Internet Archive on November 1, 2011, IBM Redbooks were highly popular, with over “500,000 downloads every month” and “80,000+ technical professionals subscribe[d] to the IBM Redbooks Weekly Newsletter” at the time. Ex-1006, 00005. The IBM Redbooks website was, and still is, searchable by keyword. *See* Ex-1006, 00007.

48. Williams discloses well-known techniques for “using Internet protocols and standards.” Ex-1005, §1 at 00025. In particular, Williams discusses the use of “encodings” in “distributed computing environments.” Ex-1005, §1.4.3 at 00039. Williams explains that the steps “translat[ing] to and from a protocol format” were known as “*marshalling* and *unmarshalling*.” *Id.* (emphasis in original).

49. Williams is analogous art because it is in the same field of endeavor as the ’693 patent, which is data processing and analysis. *Supra* §III. Williams discusses technologies related to “technologies that support the connecting or

sharing of resources and data in a very flexible and standardized manner.” Ex-1005, §1.1 at 00026. Williams is reasonably pertinent to the problem addressed by the ’693 patent’s inventor, because it teaches various techniques related to “distributed computing” (Ex-1005, §1.2.2 at 00028), which the ’693 patent states were commonly used for implementing “data processing and analysis” (Ex-1001, 1:18-21).

3. Independent Claim 1

50. It is my opinion that a POSITA would have found claim 1 rendered obvious by Foster alone, and Foster in view of Williams.

[1pre] A method comprising:

51. I understand that a preamble generally does not limit a claim.

Regardless, it is my opinion that Foster teaches the preamble.

52. Foster teaches a **method** for processing real-time and slow-moving data on a distributed computing platform. For example, Foster discloses “a Big Data platform, based on two products: IBM InfoSphere Streams and IBM InfoSphere BigInsights” and “describe[s] various scenarios where data analysis can be performed across the two platforms to address the Big Data challenges.” Ex-1004, §1.1.4 at 00037. As part of the operations of these platforms, Foster describes methods for the “[t]he integration of data-in-motion (InfoSphere Streams) and data-at-rest (BigInsights) platforms [that] addresses three main

application scenarios.” Ex-1004, §7.1.1 at 00338-339. The “scalable data ingest,” “bootstrap and enrichment,” and “adaptive analytics” scenarios encompass a number of functions and operations performed by Streams and BigInsights, such as processing data, marshalling data, updating models, and transferring data as I explain below for limitations [1a]-[1e].

53. For at least these reasons, the combination of Foster and Williams also teaches the preamble.

[1a] determining via a processor at least one processing element of at least one dynamic processing mechanism;
--

54. It is my opinion that Foster renders obvious this limitation.

55. Foster teaches **determining...at least one processing element of at least one dynamic processing mechanism**. For example, Foster discloses “scalable data ingest” processing by Streams that determines data from input streams to be further processed. Ex-1004, §7.1.1 at 00338, §7.1.2 at 00339-340 (“Data from various systems arrives continuously, that is as a continuous stream.... Data needs to first be processed for extracting all the required data for consumption by downstream analytics.”):

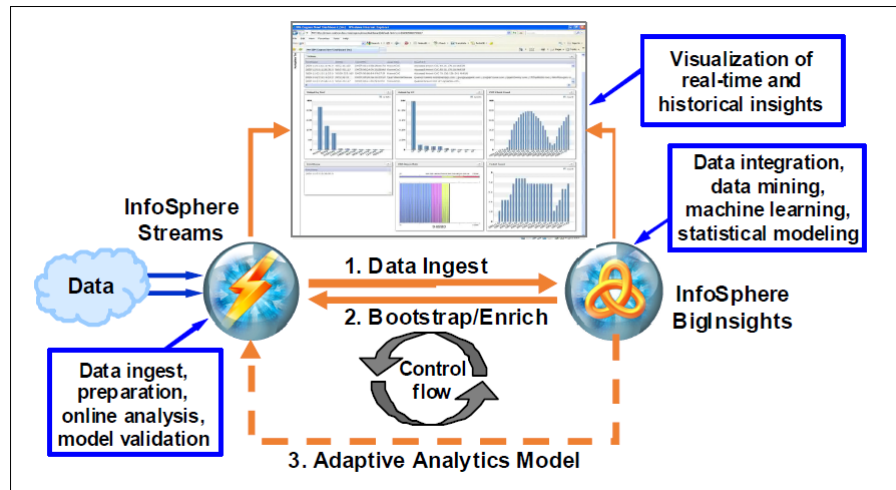


Figure 7-1 Big Data application scenarios

Fig. 7-1 at 00339 (“Data Ingest”).

56. The data determined to be sent from Streams to BigInsights is a **processing element** in the context of the ’693 patent, because it is earmarked to be further processed (e.g., bootstrapped and enriched) at BigInsights. Ex-1004, §7.1.3 at 00340 (“BigInsights can be used to analyze data.... Results from this analysis...are also used to enrich incoming data with additional attributes required for downstream analytics.”). As the ’693 patent states, a PE “may include computational code and/or data.” Ex-1001, 6:63-7:6; *supra* §IV.A.

57. Streams, including its data ingest function, is a **dynamic processing mechanism**, because Foster discloses it is a mechanism for processing data in real-time or near real-time. *See supra* §IV.B. For example, Foster discloses that Streams’ data ingestion and preparation functions “[c]ontinuous[ly] ingest” data and provide “real-time observations” to the downstream functions on BigInsights.

Ex-1004, §7.1.1 at 00338-339. “Streams supports...huge volumes of data...to be *analyzed and acted upon with low latency in close to or in real time.*” Ex-1004, §2.1 at 00056-57:

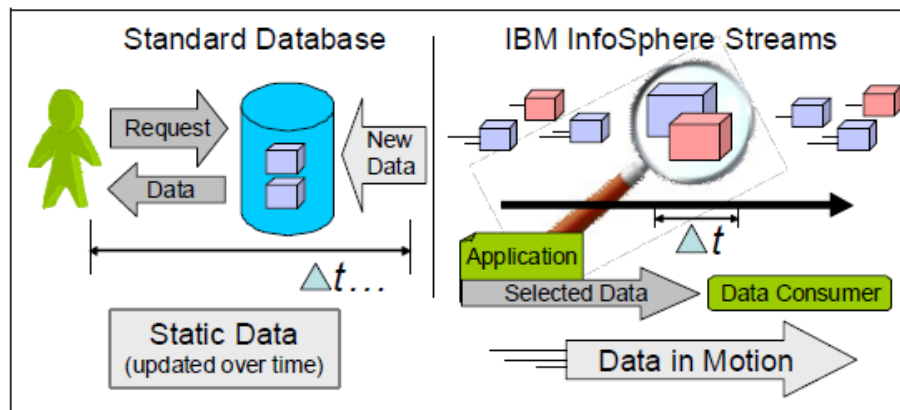


Figure 2-2 A standard relational database compared to IBM InfoSphere Streams

Fig. 2-2 at 00058 (“Data in Motion”).

58. Indeed, Foster explains “Streams supports a widely *dynamic* data model and data,” where “the data of interest within the stream is defined, and then *processed as it flows by* the application.” Ex-1004, §2.1 at 00058. Streams’ real-time data ingestion contributes to “the seamless functioning of data-in-motion (current data) and data-at-rest (historical data) analysis.” Ex-1004, §7.1 at 00338, §7.14 at 000341 (same). This is reflected in the reports presented to users, as shown in Figure 7-1:

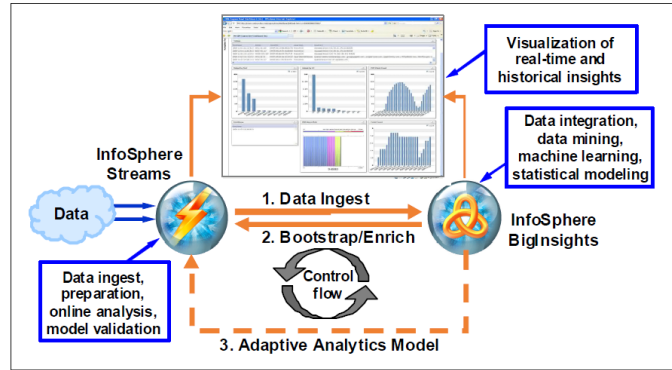


Figure 7-1 Big Data application scenarios

Fig. 7-1 at 00339 (“Visualization of real-time...insights”).

59. Foster teaches the PE (data) is **determin[ed] via a processor** because it discloses that Streams is an application run on a processor. For example, Foster discloses that Streams instances “execut[e] across one or more host computers.” Ex-1004, §2.2.1 at 00060-64 (“An Application host is dedicated to executing the actual Streams applications that were previously described as being the queries that run continuously inside Streams.”). Figure 2-3 shows how the Streams instance is comprised of Streams applications running on hosts.

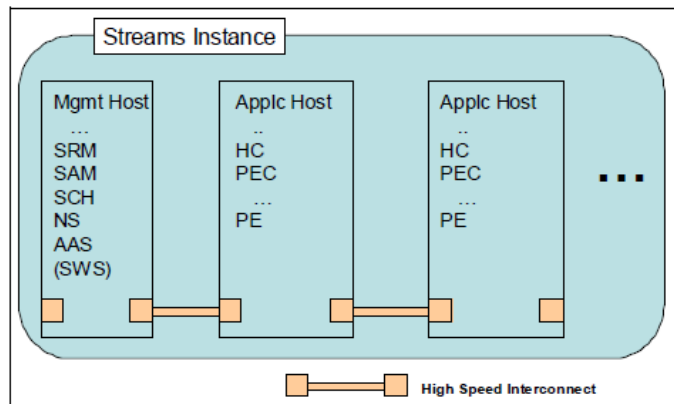


Figure 2-3 Streams instance, hosts, and more

Ex-1004, Fig. 2-3 at 00041. It was well-known to carry out processing on such host computers with processors. For example, Foster teaches using an x86 architecture processor. Ex-1004, 000050 (“IBM developed an InfoSphere Streams based trading prototype *running on a single 16 core x86 computer...*”); *see also*, Ex-1004, Fig. 1-8 (“x86 multicore”) at 00040:

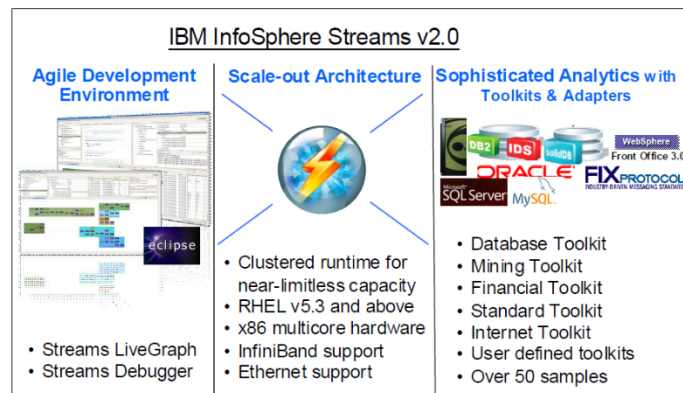


Figure 1-8 The components of InfoSphere Streams V2.0

60. Foster further explains that Streams applications are a “collection of operators” that “defines how the run time should analyze a set of stream data.” Ex-1004, §2.2.2 at 00064. This includes the processing for data ingestion and preparation. Ex-1004, §7.1.6 at 00343 (“Streams processes data using a flow graph of interconnected operators.”), §7.1.4 at 00341 (“operators on Streams continue to observe incoming data”). “An operator is a component of processing functionality that takes one or more input streams as input, processes the tuples and attributes [of the input stream], and produces one or more streams as output.” Ex-1004, §2.2.2 at 00064. The operators are assigned to processors on host computers to carry out their processing. *See, e.g.*, Ex-1004, §3.2.4 at 00117 (“By assigning

the operator's threads to different processors or hosts, consecutive tuples may be processed concurrently.”). Indeed, Foster discloses the typical environment to execute Streams to be “processor[s] in [] clusters” that can be managed through the monitoring functions of the Streams Runtime. Ex-1004, §1.2.1 at 00043.

61. For at least these reasons, the combination of Foster and Williams also teaches limitation [1a].

[1b] causing, at least in part, a marshallng via the processor of the at least one processing element as at least one data object;

62. It is my opinion that Foster alone, and Foster combined with Williams renders this limitation obvious.

Foster teaches this limitation.

63. Foster teaches **causing a marshallng...of the at least one processing element as at least one data object**. For example, Foster discloses that Streams ingests and prepares the incoming data for further processing (e.g., enriching) by BigInsights. Ex-1004, §7.1.2 at 00319 (“The parallel pipeline architecture of Streams is used to batch and buffer data and, in parallel, load it into BigInsights....”). *First*, Streams marshals the input data by processing it in various methods to convert its memory representation into a form that BigInsights can use:

Data needs to first be processed for extracting all the required data for consumption by downstream analytics. Data-preparation steps include

operations such as data-cleansing, filtering, feature extraction, deduplication, and normalization.

Ex-1004, §7.1.2 at 00319.

64. Foster illustrates this process with two examples of preparing a call detail record (CDR) and preparing a social media post:

An example of this function is clear in the Call Detail Record (CDR) processing use case. CDRs come in from the telecommunications network switches periodically as batches of files. Each of these files contains records that pertain to operations such as call initiation, and call termination for telephones. It is most efficient to *remove the duplicate records in this data as it is being ingested*, because duplicate records can be a significant fraction of the data that will needlessly consume resources if post-processed. Additionally, *telephone numbers in the CDRs need to be normalized and data appropriately prepared to be ingested into the back end for analysis*. These functions are most efficiently performed using Streams.

Another example can be seen in a social media based lead-generation application. In this application, *unstructured text data from sources such as Twitter and Facebook is ingested to extract sentiment and leads of various kinds*. In this case, a lot of resource savings can be achieved if the text extraction is done on data as it is being ingested and irrelevant data such as spam is eliminated. With volumes of 140 million tweets every day and growing, the storage requirements can add up quickly.

Ex-1004, §7.1.2 at 00340.

65. *Second*, Foster discloses details of marshalling data from the memory representation in Streams to the format for BigInsights. Ex-1004, §7.1.7 at 000344 (“Streams formatting operators can transform data between the Streams tuple format and data formats used by BigInsights.”). Streams receives streams of data records that are represented in memory as “tuples, which are composed of a fixed set of attributes.” Ex-1004, §2.2.2 at 00064. These tuples are transformed, for example, to “JSON as the data format for storage in BigInsights.” *Id.* “The TupleToJSON operator is used to convert the tuples in Streams to a JSON string for storage in BigInsights.” *Id.* Conversely, when data is sent from BigInsights to Streams, it is marshalled again using the inverse operation. *Id.* (“The JSONToTuple operator is used to convert the JSON string read from BigInsights to a tuple for Streams to process.”). Foster discloses Stream supports various other “Sink operators” to marshal data into other output formats. Ex-1004, §3.1.3 at 00095-97 (e.g., data streams for Microsoft ODBC, or Apache HDFS); *see also* Ex-1004, §7.1.6 at 00343 (“The data ingest is achieved using a Streams-BigInsights Sink operator to write to BigInsights (refer to 7.1.7, “Enabling components” on page 324).”), §7.1.7 at 00346 (“Streams source and sink adapters can be used to exchange data with BigInsights.”).

66. The resulting transformed output is a **data object**. As I just explained, Foster transforms the memory representation of the data into a format

suitable for transmission and processing at BigInsights. It is an object representing the instance of data in a particular format (e.g., JSON). This is commensurate with the '693 patent's disclosure that "marshalling may be considered as a process for transforming memory representation of an object *to a data format...suitable for...transmission of the data to different components of a process and/or from one process to another process.*" Ex-1001 ('693 patent), 4:29-47. A POSITA would have understood that a data object is simply a piece of information packaged in a specific manner, e.g., resulting from a marshalling process and is in a format suitable for use or transmission.

67. JSON was a well-known and common data format. In addition to being, and because it was, suitable for storage, many database and cloud computing systems supported JSON both for storage and as a working form for data objects. For interoperability purposes, it was commonplace to marshal data and information into JSON to transfer data between different systems and platforms, such as IBM's Streams/BigInsights platforms. *Supra* §I.C.

68. Foster teaches this marshalling occurs **via the processor**. Foster discloses the data-preparation and conversion steps are processed by operators on Streams. Ex-1004, §7.1.6-7 at 00343-344, §3.1.3 at 00095-97. As I explained above for limitation [1a], Streams operators execute on processors in host

computers. Ex-1004, §2.2.1-2.2.2 at 00060-64, §3.2.4 at 00117, §1.2.1 at 00043; *supra* §V.A.3 (incorporated by reference).

69. A POSITA would have understood the Sink operators (including TupletoJSON) to be executed on the same processor as the data-ingestion and -preparation operators, because it was conventional to use a single processor to carry out related functions on the same data. It would have been inefficient and wasteful to transfer data to another processor in the cluster. Foster aims to “optimize throughput, latency and scalability” of the data analytics carried out by Streams applications. Ex-1004, §4.5 at 00199-201. A POSITA would have realized that unnecessarily transmitting data across different nodes could increase network load and processing times. Thus, a POSITA would have found it obvious for these related operations to be executed on the same processor.

70. Additionally, Foster discloses sending information between Streams and downstream systems over a TCP network connection. For example, Foster discloses that, for “interact[ing] with systems that consume the output of a Streams application,” a Sink operator “can output a data stream over a TCP network connection, providing a reliable communication of the stream contents to the recipient.” Ex-1004, §3.1.3 at 00095. BigInsights is one such recipient of data transferred over the network from Streams. *See, e.g.*, Ex-1004, §7.1.1 at 00338 (data ingest), Fig. 7-1 at 00339 (same), §7.1.4 at 00341 (“new information” from

Streams), §7.1.5 at 00342-343 (Streams “requests an update from BigInsights”), §7.1.6 (“trigger,” “query parameters”), Fig. 7-2 at 00342 (arrows from Streams to BigInsights). A POSITA would have understood that Foster’s disclosure of Streams sending information to BigInsights via, e.g., TCP connection teaches marshalling the data before it is sent over the connection because it was well-known that sending data via TCP involves marshalling the data into the TCP packet format.

Williams further teaches this limitation.

71. Well before the ’693 patent was filed, it was customary to marshal data/information for transmission between nodes in a distributed computing environment; indeed, marshalling was a well-known and pervasive concept. *Supra* §I.C.

72. To the extent Patent Owner may argue that Foster does not explicitly disclose limitation [1b], a POSITA would have found this limitation obvious based upon the combined teachings of Foster and Williams. For example, Williams discloses, “marshalling and unmarshalling” of data values that are “translated to and from a protocol format.” Ex-1005, 00039 (emphasis removed). A POSITA would have been motivated to combine this teaching with Foster’s teachings as I explain in more detail further below.

73. Williams teaches **marshalling a processing element** (e.g., data, code, or both) **as a data object**. For example, Williams discloses that “[i]n distributed computing environments,² *encodings* define how data values defined in the application can be translated to and from a protocol format. We refer to these translation steps as *serialization* and *deserialization*, or, synonymously, *marshalling* and *unmarshalling*.” Ex-1005, 00039 (original emphasis in italics, added emphasis underlined). The ’693 patent shares this understanding, providing serializing and encoding as examples of marshalling. Ex-1001, 8:3-13 (“For example, in a distributed computing environment, an object is marshalled (e.g., encoded, serialized) for sending the object from a client to a server side or from one process to another, and unmarshalling is a process to decode (e.g., de-serialize) the encoded object at the receiving side.”), 7:20-24 (“[M]arshalling is a process of encoding an object for sharing, sending, and/or communicating the object....”).

74. Moreover, Williams discusses marshalling processing elements (e.g., code, data, or both) as data objects at least because it discusses marshalling/encoding “data values.” Ex-1005, 00039. This is commensurate with

² A POSITA would have understood that Williams’ reference to distributed computing environments encompass Foster’s Streams and BigInsights.

the '693 patent's description of data objects as being the output of a marshalling operation applied to data. Ex-1001 ('693 patent), 11:33-40, 12:40-52.

75. Thus, a POSITA would have found it obvious to take Foster's data processed by Streams (**processing elements** discussed above for limitation [1a]) and marshal those teachings with the teachings of Williams into a portable form as **at least one data object** to be transferred to and used by other processes or applications (e.g., BigInsights).

Motivation to Combine

76. A POSITA would have found the combination obvious and would have been motivated to combine Foster and Williams for a number of reasons.

77. *First*, the teachings of the references themselves strongly suggest combining their teachings in the manner I describe herein. For example, Foster discloses transferring data between applications/platforms (e.g., Streams and BigInsights) in a distributed computing environment, marshalling data for transmission between applications/platforms in a distributed computing environment was customary and well-known, and a POSITA would have understood Foster to teach and Williams to disclose marshalling data for transmission between applications/processes. *Supra* §I.C. Therefore, a POSITA would have been motivated to combine the two references.

78. *Second*, such a combination of prior art elements according to known methods would have yielded predictable results. Foster teaches transferring data between the Streams and BigInsights applications in a distributed computing environment. *See infra* §V.A.3.[1c]-[1e]. A POSITA would have understood Williams to teach marshalling data for purposes of communicating/transmitting the data between different processes in a distributed computing environment (e.g., by formatting to a protocol). The combination yields the predictable results of applying Williams’ teachings regarding marshalling data for transmission between two different applications/processes, to Foster’s teachings of transferring data between two different applications/processes (e.g., Streams and BigInsights) in order to marshal such data to put it into a format suitable for transmission, which was the known purpose of marshalling as Williams discloses.

79. *Third*, Foster and Williams, particularly with respect to the teachings I describe herein, are analogous art at least because they are applied to similar applications in a similar environment—distributed computing. Ex-1004, §1.1.4 at 00037 (“[A] Big Data platform, based on two products [that] are built to run on large-scale, *distributed* systems....”; Ex-1005, 00039 (“In distributed computing environments, *encodings* define how data values defined in the application can be translated to and from a protocol format. We refer to these translation steps as *serialization* and *deserialization*, or, synonymously, *marshalling* and

unmarshalling.”) (original emphasis in italics, added emphasis underlined).

Moreover, both references were published as part of the IBM Redbook series, which was a very popular source of information for engineers working in the field.

80. A POSITA would have looked to well-known marshalling schemes, such as Williams’ teachings, to implement data transfer between Streams and BigInsights according to Foster’s teachings. Both references teach transferring data between applications/processes and Williams teaches that such transferring involves marshalling the data according to well-known and long-established techniques (*supra* §I.C): Foster teaches transferring data via protocol formats such as JSON or TCP, as explained above, and Williams teaches that marshalling involves translating data “to and from a protocol format,” including encoding and serialization Ex-1004, §3.1.3 at 00095, §7.1.7 at 00344; Ex-1005, 00039. Given how well-known marshalling was, applying Williams’ marshalling teachings to Foster’s transfer of data between the Streams and BigInsights applications would have yielded extremely predictable results—the data would have been marshalled for transmission.

81. *Fourth*, the combination would have also involved applying a known technique (marshalling) to a known device/method (distributed computing system with dynamic and static processing mechanisms, *see supra* §I.A) ready for improvement to yield predictable results (*see supra* ¶¶78, ¶80). Williams’

marshalling was a known technique for processing data for transmission between different processes/applications.

82. A POSITA would have realized that Foster's distributed computing system was ready for improvement because Foster already contemplated formatting data for transmission to and use by downstream applications, and this formatting would have benefitted from the known implementation details regarding marshalling as taught by, for example, Williams. Implementing marshalling for Foster's transfer of data between applications using known marshalling techniques would have been straight-forward because marshalling data was ubiquitous in distributed computing and data transmission applications at the time, as Williams and other references teach. Ex-1005, §1.4.3 at 00039; *supra* §I.C.

83. Combining Foster's and Williams' teachings would not have changed either reference's principle of operation because the combination simply uses teachings from each reference according to their known principles, in the manner taught by each reference. Foster's teachings of formatting data for transmission between applications/processes, when combined with William's related teachings of marshalling in a distributed computing environment, would have still been applied in the manner taught by both references, just with straightforward and well-known implementation details regarding marshalling disclosed by Williams.

Implementing Foster's formatting and transmission teachings using Williams' marshalling teachings would have involved a simple matter of applying known marshalling techniques, which are fundamentally mathematical algorithms that process data, and as such, can be easily implemented as computer software programs running on a computer or processor(s) such as Foster's operators/processors.

84. A POSITA would have had a reasonable expectation of success when combining Foster's and Williams' teachings in the manner I describe herein because they are combined in a conventional manner that does not change any principle of operation but instead simply involves tweaks or the addition of implementation details. The combination is based on compatible devices and techniques in data processing/formatting. Foster's and Williams' teachings, when combined, would each have been used for their known purposes in the conventional manner taught by each reference. A POSITA would have been more than capable of combining the two references because the combination would simply marshal data being transferred between Streams and BigInsights according to known techniques.

85. A POSITA would have been fully capable of combining Foster's and Williams' teachings, including making necessary modifications—if any—that, as explained above, would have involved straightforward processing of information

using mathematically defined algorithms and implementing processes with straightforward programming. Marshalling and the reasons for doing it were very well-known in the prior art, as I explained above. *Supra* §I.C.

86. Based on my education and experience in the field of distributed computing and data processing (*infra* §VI), including as a graduate student, overseeing graduate students, and working with researchers in the field, it is my opinion that developing software and hardware for distributed computing and marshalling data was a common task for artisans in the field; and such development routinely involved combining and modifying different data formatting and transmission techniques and technologies. Thus, it is my view that a POSITA would have been able to combine and would have combined the teachings of Foster and Williams discussed above in the manner I describe herein. *See also supra* §III.

87. For at least these reasons, Foster alone, and Foster in combination with Williams also teaches limitation [1b].

[1c] transferring of the at least one data object to at least one static processing mechanism;

88. It is my opinion that Foster renders this limitation obvious.

89. Foster teaches **transferring of the at least one data object to at least one static processing mechanism**. For example, Foster discloses that the ingested

data marshalled into a data object (e.g., in the JSON or TCP formats) is transferred from Streams to BigInsights to further process the data. Ex-1004, §7.1.1 at 00339 (“Historical context generated from BigInsights to bootstrap analytics and enrich incoming data on Streams...Models generated by analytics such as data-mining, machine-learning, or statistical-modeling on BigInsights”). This includes the ingested data from Foster’s examples of CDR and social media feed that were marshalled at Streams, as I explained above for limitation [1b]. *Supra* §V.A.3.[1b]. Figure 7-1 illustrates this “data flow” from Streams to BigInsights. Ex-1004, §7.1.4 at 00341, Fig. 7-1 at 00339 (Data Ingest arrow from Streams to BigInsights):

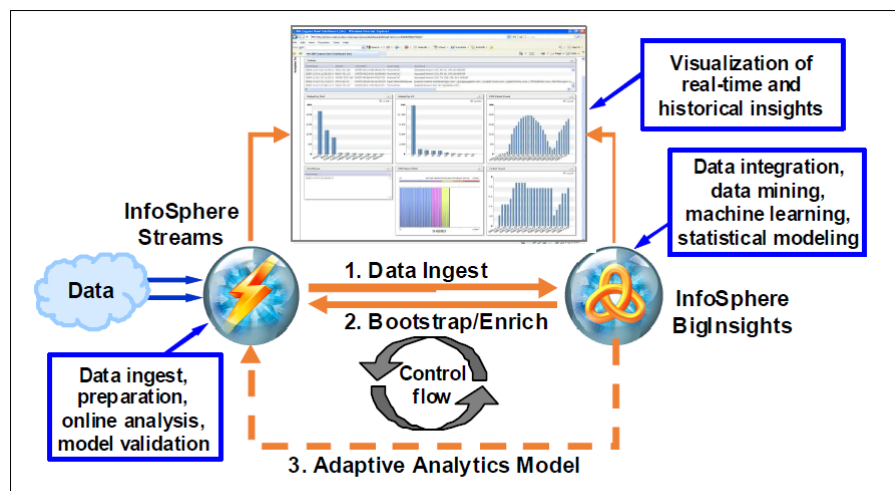


Figure 7-1 Big Data application scenarios

See also Ex-1004, §7.1.1 at 00338 (“[c]ontinuous ingest of data through Streams into BigInsights.”). Thus, a POSITA would have understood these flows to indicate transfer of the ingested data to BigInsights.

90. The thus transferred data object is then processed or stored (for processing later) at BigInsights. Ex-1004, §7.1.5 at 00342 (“The input streams are also ingested into BigInsights for historical storage and deep analysis”), §7.1.2 at 00339 (“Data is then stored in BigInsights for deep analysis....”). Foster thus confirms the transfer from Streams to BigInsights, because data arriving at Streams would have to be transferred to BigInsights if it were to be stored or processed by BigInsights.

91. BigInsights, including its deep analysis functions, is a **static processing mechanism**, because Foster teaches it is a mechanism for processing historical or slow-moving data. *See supra* §IV.C. For example, Foster discloses that BigInsights is a “data-at-rest [] platform[.]” Ex-1004, §7.1.1 at 00338. BigInsights generates “[h]istorical context” and carries out analytics such as “data-mining, machine-learning, or statistical-modeling.” Ex-1004, §7.1.1 at 00339, Fig. 7-1 at 00339 (“Data integration, data mining, machine learning, statistical modeling”). “BigInsights can be used to analyze data over a large time window, which it has assimilated and integrated from various continuous and static data sources.” Ex-1004, §7.1.3 at 00340.

92. Data-mining, machine-learning, and statistical-modeling are examples of data processing techniques that are typically carried out on large sets of

historical or slow-moving data. Indeed, Foster discloses that BigInsights does the same:

Deep analysis is performed using BigInsights to detect patterns on data collected over a long period of time. Statistical analysis algorithms or machine-learning algorithms are compute-intensive and run on the entire historical data set, in many cases making multiple passes over the data set, to generate models to represent the observations.

Ex-1004, §7.1.4 at 00341. Among other techniques, BigInsights uses the “MapReduce framework.” Ex-1004, §7.1 at 00338. MapReduce was a well-known process carried out in batches—that is, slow-moving data that is collected until it is “periodically” processed. *Supra* §I.B. The ’693 patent acknowledges that “map-reduce batch jobs” are an example of “static processing.” Ex-1001, 5:4-11.

93. For at least these reasons, the combination of Foster and Williams also teaches limitation [1c].

[1d] processing of the at least one data object by the at least one static processing mechanism; and

94. It is my opinion that Foster renders this limitation obvious. I explained above how Foster alone, and combined with Williams, teaches PEs (data) from input streams are determined and marshalled as data objects (ingested, prepared, and transformed) and transferred to BigInsights. *Supra* §V.A.3.[1a]-[1c].

95. Foster teaches **processing of the at least one data object by the at least one static processing mechanism**. For example, BigInsights bootstraps and enriches the incoming data. Ex-1004, §7.1.3 at 00340. Foster explains how BigInsights uses historical data it has accumulated and analyzed to “bootstrap [data] to a well-known state.” *Id.* The results from analyzing historical data “are also used to enrich incoming data with additional attributes required for downstream analytics.” *Id.* As I explained above for limitations [1b]-[1c], Foster provided examples of marshalling CDR and social media feed into data objects that were transferred to BigInsights. *Supra* §V.A.3.[1b]-[1c]. Following up on these examples, Foster explains that BigInsights further processes these data objects:

As an example from the CDR processing use case, an incoming CDR may only list the phone number to which that record pertains. However, a downstream analytic may want access to all phone numbers a person has ever used. *At this point, attributes from historical data are used to enrich the incoming data to provide all the phone numbers. Similarly, deep analysis results in information about the likelihood that this person may cancel their service.* Having this information enables an analytic to offer a promotion online to keep the customer from leaving the network.

In the example of the social media based application, an incoming Twitter message only has the ID of the person posting the message. However, *historical data can augment that information with attributes*

such as influencer, giving an opportunity for a downstream analytic to treat the sentiment expressed by this user appropriately.

Ex-1004, §7.1.3 at 00340.

96. *Additionally and in the alternative*, Foster teaches this limitation with its disclosure of BigInsights’ processing the transferred PE data object to generate and/or update an “adaptive analytics model” that provides the control flow for Streams’ ingestion of data. Ex-1004, §7.1.1 at 00339 (“Adaptive analytics model: *Models generated by analytics such as data-mining, machine-learning, or statistical-modeling on BigInsights* used as basis for analytics on incoming data on Streams and updated based on real-time observations.”), Fig. 7-1 (“3. Adaptive Analytics Model”):

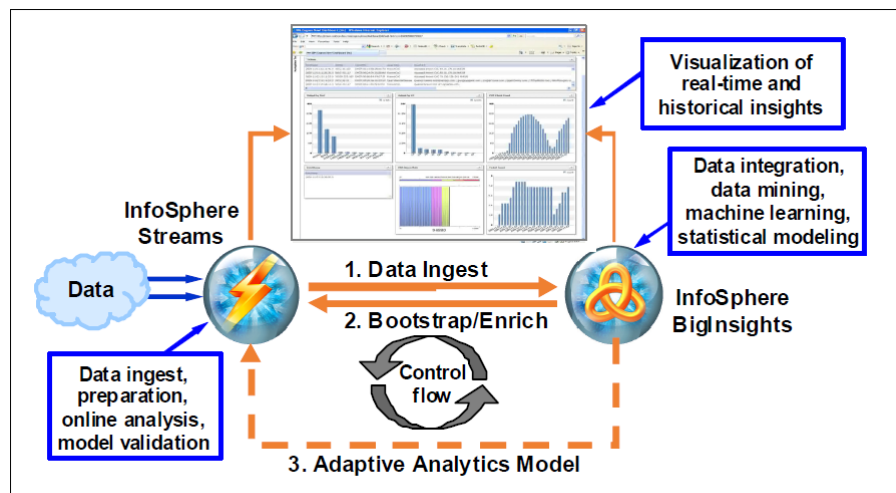


Figure 7-1 Big Data application scenarios

97. Foster further explains how BigInsights updates the model with the transferred data object, e.g., “new information of interest [] identified” by and transferred from Streams. Ex-1004, §7.1.4 at 00341 (“Where entirely new

information of interest is identified [by Streams], the deep analysis [at BigInsights] may be targeted to just update the model in relation to that new information[.]”). Specifically, “BigInsights does deep analysis using the same sentiment extraction analytic as used in Streams and creates a results file to update the Streams model.” Ex-1004, 7.1.6 at 343. Foster illustrates parts of this process in relation to a limited example that monitors for positive/negative sentiment of social media feed. BigInsights “queries all of its data using the same sentiment analytics used in Streams and recalculates the list of known causes” that Stream uses to monitor the incoming social media feed. Ex-1004, §7.1.5, Fig. 7-2 (“Re-calculate watch list of known causes”):

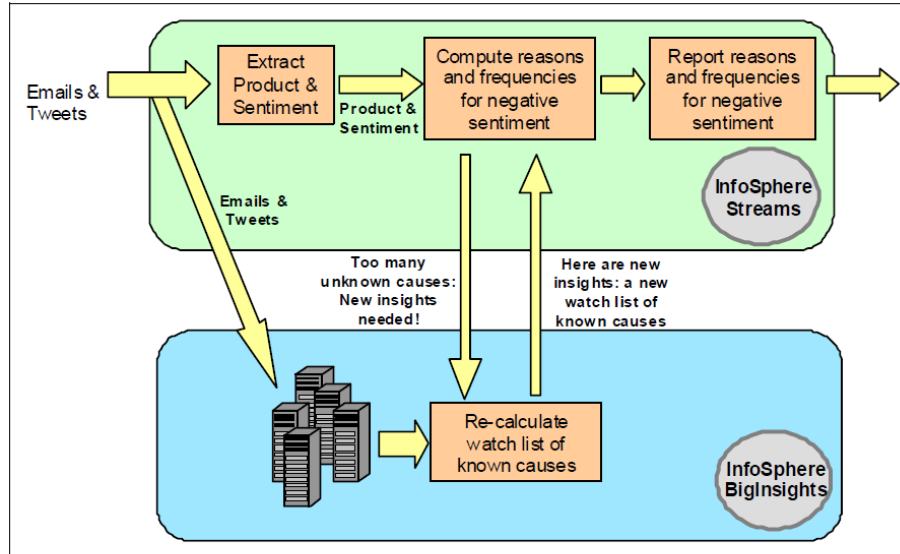


Figure 7-2 Streams and BigInsights application example

98. I already explained above how BigInsights is a **static processing mechanism**. *Supra* §V.A.3.[1c] (analysis incorporated here by reference).

99. For at least these reasons, the combination of Foster and Williams also teaches limitation [1d].

[1e] after processing, transferring the at least one data object back to the at least one dynamic processing mechanism.

100. It is my opinion that Foster renders this limitation obvious.

101. I explained above how Foster alone, and combined with Williams, teaches PEs (data) from input streams are determined, marshalled as data objects (ingested, prepared, and transformed), transferred to BigInsights and is then processed by BigInsights. *Supra* §V.A.3.[1a]-[1d].

102. Foster teaches **after processing** at BigInsights (the static processing mechanism), **transferring the at least one data object back to the at least one dynamic processing mechanism**. For example, Foster discloses the bootstrapped or enriched data is used in “downstream analytics.” Ex-1004, §7.1.3 at 000340. Such downstream analytics include “downstream analytics on Streams.” Ex-1004, §7.1.2 at 00339. Figure 7-1 confirms that the bootstrapped/enriched data is transferred back to Streams with its arrow from BigInsights to Streams showing the data flow for “2. Bootstrap/Enrich”:

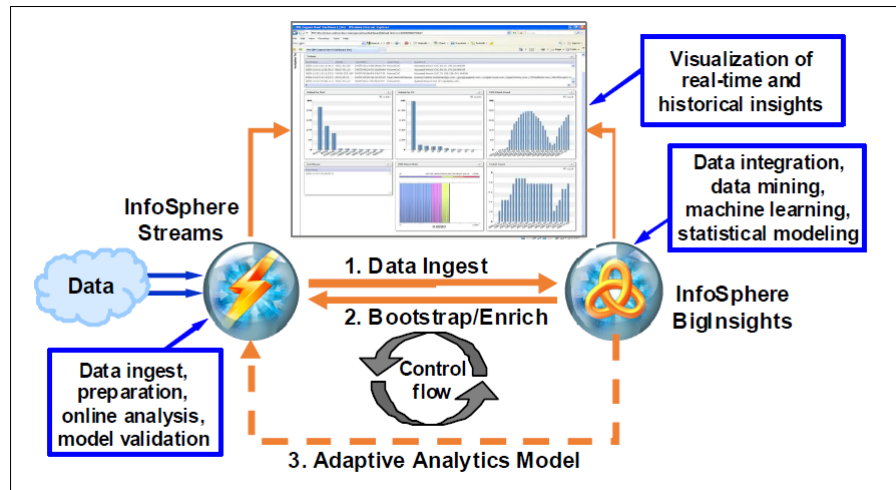


Figure 7-1 Big Data application scenarios

Ex-1004, Fig 7-1 at 00339.

103. For the CDR processing example, Foster explains that “a downstream analytic may want access to all phone numbers a person has ever used” or that BigInsights generated “information about the likelihood that this person may cancel their service... enables an analytic to offer a promotion....” Ex-1004, §7.1.3 at 00340. For the social media example, BigInsight can “can augment that information with attributes such as influencer, giving an opportunity for a downstream analytic to treat the sentiment expressed by this user appropriately.” *Id.*

104. I already explained above how Streams is a **dynamic processing mechanism**. *Supra* §V.A.3.[1a] (analysis incorporated here by reference).

105. For at least these reasons, the combination of Foster and Williams also teaches limitation [1e].

4. Independent Claim 11

106. Claim 1 is directed to a method where certain steps are performed “via a processor” or by a “processing mechanism.” Claim 11 is directed to an apparatus that includes a processor and a memory that contains computer program code configured to perform the steps of claim 1’s method, where some steps are performed via a processor or by one or more of the apparatus with a processor and a “processing mechanism.”

107. The following chart shows that the claims are substantially the same, except that claim 11 further requires “at least one processor” and “at least one memory including computer code for one or more programs, the at least one memory and the computer program code configured to, with the at least one processor, cause the apparatus to perform” the same method limitations of claim 1. Only the differences between the two claims are bolded and colored green below:

[1pre] A method , comprising:	[11pre] An apparatus , comprising:
	[11a] at least one processor; and at least one memory including computer program code for one or more programs, the at least one memory and the computer program code configured to, with the at least one processor,

	cause the apparatus to perform at least the following,
[1a] determining via a processor at least one processing element of at least one dynamic processing mechanism;	[11b] determine at least one processing element of at least one dynamic processing mechanism;
[1b] causing, at least in part , a marshalling via the processor of the at least one processing element as at least one data object;	[11c] cause a marshalling via the processor of the at least one processing element as at least one data object;
[1c] transferring of the at least one data object to at least one static processing mechanism; and	[11d] transfer of the at least one data object to at least one static processing mechanism;
[1d] processing of the at least one data object by the at least one static processing mechanism; and	[11e] process the at least one data object by the at least one static processing mechanism; and
[1e] after processing, transferring the at least one data object back to the at least one dynamic processing mechanism.	[11f] after the process, transfer the at least one data object back to the at least one dynamic processing mechanism.

108. As I explained above for limitation [1b], Foster discloses a distributed computing system operating host computers, that a POSITA would have understood includes components such as a processor and memory storing computer program code, that perform the method of claim 1. Such components and their use for distributed computing applications were well-known before the '693 patent.

109. The remaining substantial limitations [11b]-[11f], which correspond to limitations [1a]-[1e], recite similar language and the combination of Foster and Williams teaches the same limitations for the same reasons that I explained above for claim 11. Therefore, I address only the differences between the two claims below.

[11pre] An apparatus comprising:

110. I understand that a preamble generally does not limit a claim. Regardless, it is my opinion that Foster teaches the preamble.

111. For example, Foster discloses that Streams and BigInsights are part of IBM's BigData platform and operate on apparatuses (e.g., distributed computing systems). Ex-1004, §1.1.4 at 00037 ("Both products are built to run on large-scale distributed systems, designed to scale from small to very large data volumes, handling both structured and unstructured data analysis."), §7.1 at 00338 (same). Foster provides an example architecture with distributed hardware elements represented as cubes:

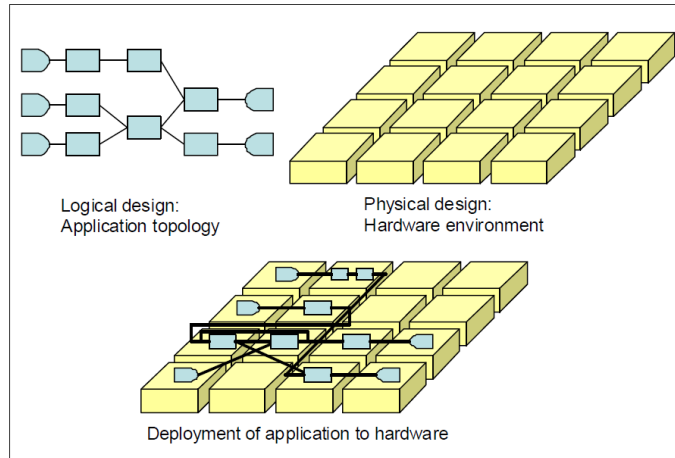


Figure 3-1 Application design, physical design, and deployment

Ex-1004, §3.1 at 00089 (“Hardware environment”).

112. The apparatus at least comprises the elements of limitation [11a] that I explain below, and performs the steps that I explain further below for limitations [11b]-[11f].

**[11a] at least one processor;
and at least one memory including computer program code for one or more programs,
the at least one memory and the computer program code configured to, with the at least one processor, cause the apparatus to perform at least the following,**

113. It is my opinion that Foster teaches this limitation.

114. Foster teaches **at least one processor**. As I explained above for limitation [1a], Foster discloses that Streams executes on host computers in data processing clusters (Ex-1004, §2.2.1 at 00060-64, §1.2.1 at 00043), and that such computers included processors (Ex-1004, Fig. 1-8 at 00040). *Supra* §V.A.3.[1a].

115. Foster teaches **at least one memory including computer program code for one or more programs**. For example, Foster discloses that Stream is a “software product.” Ex-1004, §2.2 at 00059, 00390 (“Streams software distribution file”). And “[a]ll software environments are composed of three major components: process, *memory*, and disk.” Ex-1004, §2.1 at 00059, §5.6.2 at 00287 (“A Streams Instance refers to the collection of *memory*, process, and disk that is the Streams server proper.”). The Streams runtime software can, for example, be installed on a “32-bit RHEL 5.5 VMWare image configured with 2 GB of RAM and hosted by a PC running Microsoft Windows XP.” Ex-1004, 00391.

116. The memory of Foster’s host computers and the programs in them, such as the Streams runtime and applications, include **computer program code**. Foster discloses that Streams applications are written as “[s]ource files [that] are compiled into binary code files.” Ex-1004, §5.1.1 at 00226, §5.1.3 at 00231 (“the Streams application code”), §5.3 at 00268 (“Streams code”), §6.4.2 at 00270 (Streams source file includes *code*). For example, Foster provides the source code and details of an example streams program that calls for “compil[ing] and run[ning] the software.” Ex-1004, §5.5 at 00279. A POSITA would have understood that the compiled computer program code would have been stored in the host computer’s memory **for one or more programs** constituting Streams application, as code is stored in memory in order to be executed and programs

consist of code. *See also, e.g.*, Ex-1004, §6 at 00297-336 (“Advanced Streams programming”).

117. These teachings equally apply to Foster’s teachings of BigInsights, which Foster explains is designed to run on similar “large-scale distributed systems.” Ex-1004, §1.1.4 at 00037. A POSITA would have understood these disclosures to teach that the host computers hosting BigInsights also include a **processor and memory including computer program code** for the BigInsights programs. Using such memory and code was a conventional and common approach to implement data processing in general, and distributed computing functionality in particular, and a POSITA would have understood Foster to teach these features.

118. Foster teaches **the at least one memory and the computer program code [is] configured to, with the at least one processor, cause the apparatus to perform at least the following** steps claimed in limitations [11b]-[11f], because Foster, including its teachings of Streams and Insights causing the execution of processing on distributed computing systems, teaches those limitations, as I explain below and the corresponding limitations [1a]-[1e], as I explained above. *Infra* §V.A.4.[11b]-[11f], §V.A.3.[1a]-[1e].

- | | |
|--------------|--|
| [11b] | determine at least one processing element of at least one dynamic processing mechanism; |
| [11c] | cause a marshalling via the processor of the at least one processing element as at least one data object; |
| [11d] | transfer of the at least one data object to at least one static processing mechanism; |
| [11e] | process the at least one data object by the at least one static processing mechanism; and |
| [11f] | after the process, transfer the at least one data object back to the at least one dynamic processing mechanism. |

119. It is my opinion that Foster alone, or Foster combined with Williams renders obvious these limitations [11d]-[11h]. These limitations recite substantively similar language to limitations [1a]-[1e] of claim 1. *Supra* §V.A.4. Foster alone, and Foster combined with Williams teach these limitations for the same reasons, and that analysis is incorporated here by reference. *Supra* §V.A.3.[1a]-[1e]. A POSITA would have been motivated to combine Foster and Williams for the same reasons that I explained above for limitation [1b]. *Supra* §V.A.1.[1b] (motivation to combine). For these reasons, the Foster alone, or Foster combined with Williams teaches these limitations.

5. Dependent Claims 2 and 12

120. It is my opinion that Foster alone, and Foster combined with Williams renders claims 2 and 12 obvious.

- 2[a]. The method of claim 1, further comprising: a processing of at least substantially real-time data via the at least one processing element to determine one or more triggering events,**
- 12[a]. The apparatus of claim 11, wherein the apparatus is further caused to: process and/or facilitate a processing of at least substantially real-time data via the at least one processing element to determine one or more triggering events,**

121. Foster alone and the combination of Foster and Williams each teaches claims 1 and 11, as I explained above. *Supra* §§V.A.3-4. It is my opinion that Foster teaches the additional limitation.

122. Foster teaches **a processing of at least substantially real-time data via the at least one processing element to determine one or more triggering events**. For example, Foster discloses “an additional control flow from Streams to BigInsights...in the form of a trigger” while ingesting incoming real-time data to be, or to not be processed at BigInsights. Ex-1004, §7.1.4 at 00341. As I explained above for claim 1, Foster combined with Williams teaches processing substantially real-time data (e.g, the incoming data stream) at Streams, including determining processing elements (e.g., selecting such data) for further processing (e.g., bootstrapping/enrichment) at BigInsights. *Supra* §V.A.3.[1a]-[1b].

123. During ingest, “Streams appl[ies] the [adaptive analytics] model on the incoming data” to identify relevant or remarkable input data, among other things. Ex-1004, §7.1.4 at 00341. Discussing its social media feed example, Foster discloses that during ingest, “irrelevant data such as spam is eliminated.”

Ex-1004, §7.1.2 at 00340. One way Streams monitors the incoming data is whether it has been posted by a “key influencer.” Ex-1004, §7.1.4 at 00341.

124. Foster teaches these are **triggering events** because these events (e.g., detection of an influencer’s post, absence of relevant information, and a request for further processing) trigger further processing at BigInsights, rather than discarding the data as irrelevant. Foster’s triggering applies to both the bootstrapping/enrichment during data ingest and model updating teachings, because Foster explains that Streams “triggered deep analysis in BigInsights using the same Streams-BigInsights Sink operator as mentioned in ‘Data ingest into BigInsights from Streams.’” Ex-1004, §7.1.6 at 00343.

125. *In the alternative*, in Foster’s positive/negative sentiment example that I explained above for limitation [1d], Foster teaches this limitation by disclosing another model that Stream uses to monitor for “negative sentiment” about a product. Ex-1004, §7.1.5 at 00342-343 (by using “the list of causes to be monitored in real time”), Fig. 7-2 at 000342:

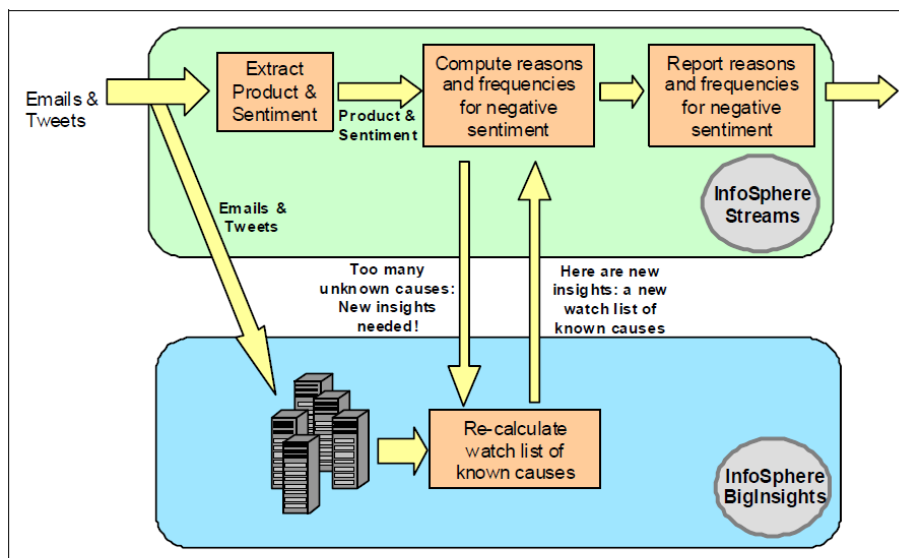


Figure 7-2 Streams and BigInsights application example

126. Among the non-discarded, relevant data selected as processing elements, Foster further processes the real-time data to determine the emergence of new trends as a **triggering event** to update the adaptive analytics model. Foster, §7.1.4 at 00341 (“If the new observations deviate significantly from the expected behavior, the online analytic on *Streams* may determine that it is time to trigger a new model-building process on BigInsights” and “[w]here entirely new information of interest is identified, the deep analysis may be targeted to just update the model in relation to that new information”). Foster discloses an example where there is a change in influencer relationships. Ex-1004, §7.1.4 at 00341 (a “key influencer identified in the model may no longer be influencing others or an entirely new influencer or relationship can be identified”). Another example is if there are “[t]oo many unknown causes” of negative sentiment. Ex-1004, §7.1.5 at 00342.

127. For at least these reasons, the combination of Foster and Williams also teaches limitations 2[a] and 12[a].

- [2b] **wherein, the transferring of the at least one data object to the at least one static processing mechanism is based, at least in part, on the one or more triggering events, and**
- [12b] **wherein the transfer of the at least one data object to the at least one static processing mechanism based, at least in part, on the one or more triggering events, and**

128. It is my opinion that Foster renders obvious this limitation.

129. Foster teaches **the transferring of the data object to the static processing mechanism is based, in part, on the triggering events**. For example, Foster discloses that incoming data that is not identified by a triggering event at Streams for further processing at ingest is discarded and thus, not transferred to BigInsights. Ex-1004, §7.1.2 at 00340 (“[I]rrelevant data such as spam is eliminated.”). The data that is relevant is ingested, marshalled, and transferred to BigInsights for bootstrap/enrichment, as I explained above for limitations [1b]-[1d]. Ex-1004, §7.1.1 at 000339-340; *supra* §V.A.3.[1b]-[1d].

130. Foster’s explanation of BigInsights’ model updating based on the triggering event of new information of interest (e.g., a new influencer) further illustrates how transferring is based on the triggering events. Foster explains that “the deep analysis may be targeted to just update the model in relation to that *new information*.” Ex-1004, §7.1.4 at 00341. In order for BigInsights “to look for all

historical context for this new influencer, where the raw data had [already] been stored in BigInsights” (*id.*), Streams would have had to transfer the new information to BigInsights, marshalled into a data object of the right format. A POSITA would have understood the transfer was based on the triggering event, because it occurred in response to the triggering event and it was intended “to trigger a new model-building process on BigInsights” (the resulting processing). *Id.*

131. *In the alternative*, Foster teaches this limitation with its positive/negative sentiment monitoring example where the transferring is based on the triggering event of identifying a significant “percentage of negative sentiment that have an unknown cause.” Ex-1004, §7.1.5 at 00342-343. The teachings that I just explained above also apply to this alternative teaching. As Foster explains, the triggering of the deep analysis is transferred to BigInsights in the same way that data is ingested into BigInsights by Stream. Ex-1004, §7.1.6 at 00343 (“BigInsights use[s] the same Streams-BigInsights Sink operator as mentioned in ‘Data ingest into BigInsights from Streams’”). It would have been obvious for Streams to transfer the data object in this way (at least including the most recent negative item that tripped the limit and the trigger to commence deep analysis), because 1) it would have been more efficient to send the two as a single data object

instead of separately; and 2) the negative item would need to be included in BigInsight's deep analysis to update the watch list of known causes.

132. For at least these reasons, the combination of Foster and Williams also teaches limitations [2b] and [12b].

[2c]	wherein the one or more triggering events include, at least in part, one or more contextual trends occurring above at least one threshold level.
[12c]	wherein the one or more triggering events include, at least in part, one or more contextual trends occurring above at least one threshold level.

133. It is my opinion that Foster renders obvious this limitation. As I just explained above for limitations [2a] and [2b], Foster teaches determining triggering events and transferring data to BigInsights (a static processing mechanism) based on triggering events such as including relevant information, identifying a deviation in expected behavior, or identifying new information of interest. Alternatively Foster teaches an example with a triggering event is an accumulation of unknown causes, as I also explained above for limitations [2a]-[2b].

134. Foster teaches that **the one or more triggering events include, in part, one or more contextual trends occurring above at least one threshold level.** For example, Foster discloses that BigInsight's model generating/updating is triggered by trends in the context of an e-mail's or tweet's relation to influencers. Ex-1004, §7.1.4 at 00341 ("a key influencer identified in the model

may no longer be influencing others or an entirely new influencer or relationship may be identified”).

135. This is similar to the ’693 patent’s example of “a contextual trend associated with, for example, *one or more users*, locations, events, topics, and the like.” Ex-1001, 8:14-34. Key influencers are users, and they are also often the topic of social media. Their presence, absence, or relation to incoming data entries (e.g., e-mail or tweets) are part of the context for the social media feed.

136. The fact that an influencer is no longer influencing others, or the emergence of a new influencer are trends (e.g., movements in data) of the context. As the ’693 patent states, “a trend may indicate a movement.” Ex-1001, 8:14-34. Foster’s triggering reflects the trend **occurring above a threshold level** of social media influence. For example, “an entirely new influencer or relationship” identifies a contextual trend occurring above a threshold level of influence (e.g., accruing a certain number of followers). Similarly, an influencer no longer influencing others identifies a trend occurring above a threshold level of losing influence. These examples demonstrate Foster’s disclosure of triggering events including contextual trends occurring above a threshold level (e.g., “new observations deviate significantly from the expected behavior” or when “the real-world has deviated sufficiently from the model’s prediction.”). Ex-1004, §7.1.4 at 00341.

137. *In the alternative*, for Foster’s positive/negative sentiment monitoring example, Foster teaches this limitation with a triggering event that is based on the contextual trend of the number of negative sentiments with unknown cause occurring above “a significant percentage.” Ex-1004, §7.1.5 at 00342. This example likewise discloses a threshold level, which is set at a numeric percentage.

138. For at least these reasons, the combination of Foster and Williams also teaches limitations [2c] and [12c].

6. Dependent Claims 3 and 13

139. It is my opinion that Foster alone, and Foster combined with Williams renders claims 3 and 13 obvious.

- 3. The method of claim 2, further comprising: unmarshalling of the at least one processing element as the at least one data object.**
- 13. The apparatus of claim 12, wherein the apparatus is further caused to: cause, at least in part, an unmarshalling of the at least one processing element as at least one data object.**

140. Foster alone and the combination of Foster and Williams each teaches claims 2 and 12, as I explained above. *Supra* §V.A.5. It is my opinion that Foster teaches the additional limitation.

141. As explained above for limitation [1b], Foster teaches marshalling data (e.g., into the TCP packet format) at Streams in order to transfer the data for use at BigInsights. Because the data has been transformed into a certain protocol format, a POSITA would have understood Foster to teach BigInsights

unmarshalling the data object (undoing the transform) in order to access the data inside for processing. *Supra* §I.C. A POSITA would also have understood from Foster that the corresponding marshalling and unmarshalling processes are applied at both Streams and BigInsights when data is transferred between the two in either direction, in order to prepare the data for transmission or for processing after transmission. *Supra* §I.C. A POSITA would have found this limitation obvious in light of Foster’s teachings, because it was customary to marshal and unmarshall data before the ’693 patent was filed. *See supra* §V.A.3.[1b]; *supra* §I.C.

142. A POSITA would have found this limitation further obvious based upon the teachings of Williams. For example, Williams discloses marshalling a processing element (e.g., data) as a data object (as I explained above for limitation [1b], *supra* §V.A.3.[1b]), and Williams further discloses unmarshalling processing elements from the data objects.³ In particular, Williams discloses “*serialization*

³ A POSITA would have understood the claim to require unmarshalling a processing element *from* a data object, because unmarshalling is the inverse process of marshalling. The ’693 patent’s disclosures related to unmarshalling are consistent with this understanding. Ex-1001, 8:3-13 (“unmarshalling process is to decode the object that was marshalled,” “an unmarshalling is a process to decode (e.g., de-serialize) the encoded object at the receiving side”), 14:12-22 (same).

and *deserialization*, or, synonymously, *marshalling* and *unmarshalling*.” Ex-1005, 00039 (emphasis original). A POSITA would have understood that a data object resulting from marshalling a processing element is unmarshalled, as taught by Williams, and the result is a processing element. *Supra* §V.A.3.[1b].

143. I explained above how and why a POSITA would have been motivated to combine these related teachings of Foster and Williams for limitation [1b]. *Supra* §V.A.3.[1b] (motivation to combine). The same reasons apply here because Foster’s and William’s teachings of unmarshalling are the flip side of the marshalling teachings.

144. For at least these reasons, the combination of Foster and Williams also teaches claims 3 and 13.

7. Dependent Claims 4 and 14

145. It is my opinion that Foster alone, and Foster combined with Williams renders claim 4 obvious.

- | |
|--|
| <p>4. The method of claim 2, wherein the processing of the at least one data object via the at least one static processing mechanism causes, at least in part, an updating of the at least one data object based, at least in part, on the one or more triggering events, the one or more contextual trends, or a combination thereof.</p> <p>14. The apparatus of claim 12, wherein the process of the at least one data object via the at least one static processing mechanism causes, at least in part, an updating of the at least one data object based, at least in part, on the one or more triggering events, the one or more contextual trends, or a combination thereof.</p> |
|--|

146. Foster alone and the combination of Foster and Williams each teaches claims 2 and 12, as I explained above. *Supra* §V.A.5. It is my opinion that Foster teaches the additional limitation.

147. As I explained above for limitation [1d], Foster teaches **the processing of the at least one data object via the at least one static processing mechanism**, by Foster's disclosures of BigInsight's bootstrapping/enrichment data received from Streams. *Supra* §V.A.3.[1d].

148. Foster further teaches **the processing...causes an updating of the data object based, at least in part, on the triggering event**. For example, as I explained above for claim 2, Foster discloses that BigInsights bootstraps/enriches the data, updating it with relevant historical information, received from Streams in response to the triggering events of 1) the data being found to be relevant (e.g., not discarded); and 2) Streams requesting enrichment or bootstrapping to Streams. Ex-1004, §7.1.4 at 00341, §7.1.2 at 00340; *supra* §§V.A.5.2[a]-[2b].

149. *In the alternative*, as I explained above for limitation [1d], Foster additionally teaches **the processing of the at least one data object via the at least one static processing mechanism** by processing the ingested data object to generate/update an adaptive analytics model that Streams uses as part of its ingestion process, discussing a limited example to monitor positive/negative sentiment in incoming social media feed. *Supra* §V.A.3.[1d].

150. This alternatively and additionally teaches **the processing...causes an updating of the data object based, at least in part, on the triggering event**. For example, Foster teaches building/updating a model at BigInsights in relation to information processed at and transferred to BigInsights from Streams, when such information triggers building/updating an adaptive analytics model in response to triggering events. Ex-1004, §7.1.4-6 at 00341-343; *supra* §§V.A.5.2[a]-[2b]. I also explained above how BigInsights' triggering events are based on contextual trends. *Supra* §V.A.5.[2c]. Therefore, the updating is also **based in part on contextual trends**.

151. Foster discloses that the information (e.g., “new information” or “new observations”) is transferred to BigInsights after being marshalled and then incorporated into an existing or new model at BigInsights, so the model will be updated “in relation to that new information.” Ex-1004, §7.1.4 at 00341, Fig. 7-1 at 00339; *supra* §V.A.3.[1d], §V.A.5.2[a]-[2b]. The data object is **updated**,

because it is incorporated into an adaptive analytics model that combines the data object with other historic information.

152. For at least these reasons, the combination of Foster and Williams also teaches claims 4 and 14.

8. Dependent Claims 5 and 15

153. It is my opinion that Foster alone, and Foster combined with Williams renders claims 5 and 15 obvious.

- 5[a]. The method of claim 4, further comprising: at least one determining of one or more computational chains, one or more parameters of the one or more computational chains, one or more data elements of the one or more computational chains, or a combination thereof associated with the at least one processing element,**
- 15[a]. The apparatus of claim 14, wherein the apparatus is further caused to: determine one or more computational chains, one or more parameters of the one or more computational chains, one or more data elements of the one or more computational chains, or a combination thereof associated with the at least one processing element,**

154. Foster alone and the combination of Foster and Williams each teaches claims 4 and 14, as I explained above. *Supra* §V.A.7. It is my opinion that Foster teaches the additional limitation. As I explained above, Foster discloses that Streams uses models and triggers BigInsights to build and update models based on data processed at and received from Streams. Ex-1004, §7.1.4 at 00341; *Supra* §V.A.3.[1d], §V.A.5.2[a]-[2c], §V.A.7.

155. Foster teaches **at least one determining of one or more computational chains** and **at least one determining of...one or more parameters of the computational chains**. For example, Foster discloses that Streams includes “query parameters to tailor the deep analysis in BigInsights” while triggering model building/updating. Ex-1004, §7.1.6 at 00343. The trigger specifies the steps of the computational chain to be carried out. For the negative sentiment monitoring example, Foster discloses that Streams triggers BigInsights to carry out deep analysis with a “sentiment extraction analytic” function. *Id.* The “query parameters” specify the parameters of the computational steps to be used by BigInsight for deep analysis. *Id.* (“For more advanced scenarios, the trigger...contain[s] query parameters to tailor the deep analysis”). Thus, the parameters determine (e.g., tailor) how the computation is to be carried out.

156. *In the alternative*, Foster teaches **determining of one or more computational chains** by building/updating adaptive analytics models. The models are computational chains, at least because using the models involves several sequential steps of computing that are used by Streams to process incoming data. For example, a model makes a prediction by applying an algorithm to a set of stored data or stored parameters. Multiple parameters and data items/elements are accessed by the model, leading to a series of steps that constitutes a chain of computations.

157. In both cases, Foster teaches the determining, the computational chains, and the parameters are **associated with the at least one processing element**. As I explained above for claim 2, the building/updating of models is triggered by Streams' analysis or processing of the processing element (incoming data). *Supra* §V.A.5.2[a]-[2c]. Therefore, both the trigger and the model are associated with the processing element.

158. For at least these reasons, the combination of Foster and Williams also teaches limitations 5[a] and 15[a].

[5b] wherein at least one data object includes, at least in part, the one or more computational chains, the one or more parameters, the one or more data elements, or a combination thereof.

[15b] wherein at least one data object includes, at least in part, the one or more computational chains, the one or more parameters, the one or more data elements, or a combination thereof.

159. It is my opinion that Foster teaches this limitation.

160. As I explained above for limitations 5[a] and 15[a], Foster discloses its triggers include computational chains and/or parameters (e.g., parameters to tailor the deep analysis). *Supra* §V.A.8.5[a]/15[a]. I also explained above for limitations [2b] and [12b], Foster teaches transferring the data object and including the trigger for model building/updating deep analysis. Ex-1004, §7.1.6 at 00342-343; *supra* §V.A.5.[2b]/[12b]. Similar reasons of efficiency would apply here and are incorporated by reference, because the trigger, the query parameters, and the

data elements are all related information that are used together by BigInsights. A POSITA would have found it obvious to marshal these together so that the **data object includes, at least in part, the computational chains, the parameters, the data elements, or a combination thereof.**

161. For at least these reasons, the combination of Foster and Williams also teaches limitations [5b] and [15b].

9. Dependent Claims 7 and 17

162. It is my opinion that Foster alone, and Foster combined with Williams renders claims 7 and 17 obvious.

- 7. The method of claim 1, wherein the processing of the at least one data object causes, at least in part, updating of one or more components of the at least one static processing mechanism.**
- 17. The apparatus of claim 11, wherein the process of the at least one data object causes, at least in part, an updating of one or more components of the at least one static processing mechanism.**

163. Foster alone and the combination of Foster and Williams each teaches claims 1 and 11, as I explained above. *Supra* §§V.A.3-4. It is my opinion that Foster teaches the additional limitation.

164. As explained above for limitations [1d], Foster teaches processing new information in data objects received from Streams to update models generated by BigInsights. Ex-1004, §7.1.4 at 00341; *supra* §V.A.3.[1d].

165. Foster further teaches the processing **causes, at least in part, updating of one or more components of the at least one static processing mechanism.** For example, Foster discloses that BigInsights’ “deep analysis may be targeted to just update the model in relation to [the] new information.” Ex-1004, §7.1.4 at 00341. The model is a component of BigInsights, maintained in BigInsights and used to generate and provide a new model for Streams. *See* Ex-1004, Fig. 7-2:

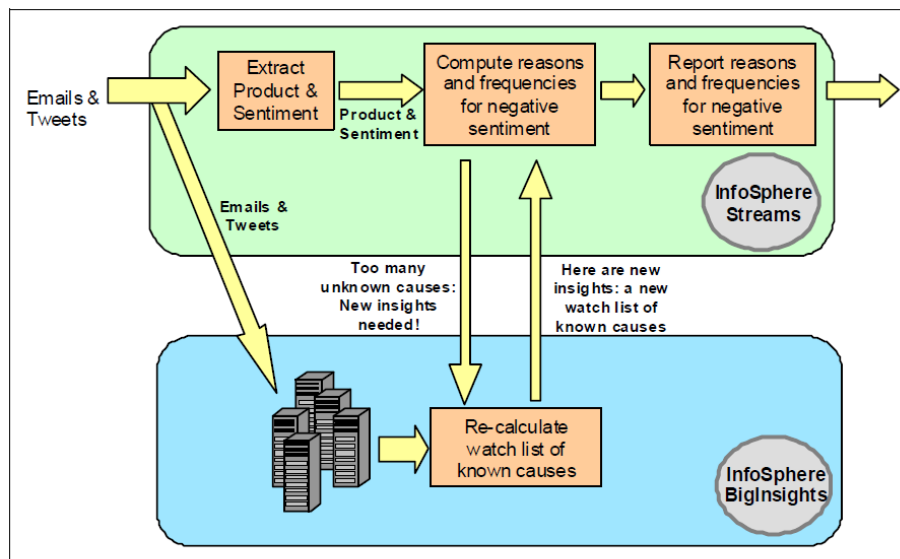


Figure 7-2 Streams and BigInsights application example

Because the updated model is used again the next time a request to update the model is triggered, a POSITA would have understood BigInsights to maintain a local copy, update it as requested, and provide the updated model to Streams.

166. For at least these reasons, the combination of Foster and Williams also teaches claims 7 and 17.

10. Dependent Claims 8 and 18

167. It is my opinion that Foster alone, and Foster combined with Williams renders claims 8 and 18 obvious.

- 8[a]. The method of claim 1, further comprising aggregating of one or more outputs of the at least one dynamic processing mechanism and the one or more other outputs of the at least one static processing mechanism; and**
- 18[a]. The apparatus of claim 11, wherein the apparatus is further caused to: cause, at least in part, an aggregation of one or more outputs of the at least one dynamic processing mechanism and the one or more other outputs of the at least one static processing mechanism; and**

168. Foster alone and the combination of Foster and Williams each teaches claims 1 and 11, as I explained above. *Supra* §V.A.3. It is my opinion that Foster teaches the additional limitation.

169. For example, Foster discloses in Figure 7-1 that the outputs of Streams (the dynamic processing mechanism) and BigInsights (the static processing mechanism) are aggregated and displayed together at a user interface in various graphs for “[v]isualization of real-time and historical insights.” Ex-1004, Fig. 7-1 at 00339:

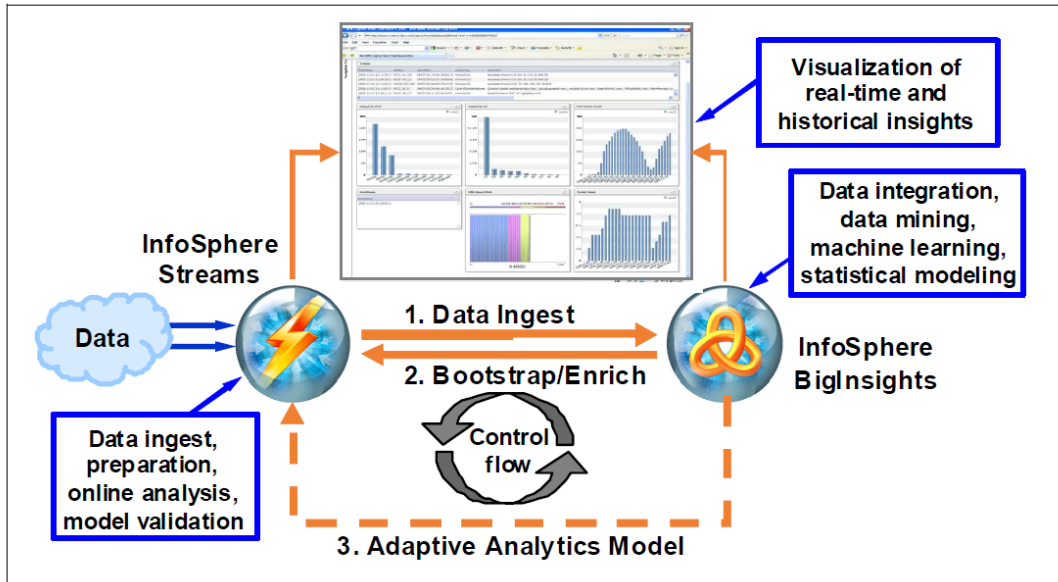


Figure 7-1 Big Data application scenarios

170. For example, the visualized data includes the enriched data that I explained above for limitation [1d]. *Supra* ¶V.A.3.[1d]. The enriched data aggregates the analysis of both Streams and Insights. Ex-1004, §7.1.3 at 00340 (“attributes from historical data are used to enrich the incoming data” and “historical data can augment that information”).

171. For at least these reasons, the combination of Foster and Williams also teaches limitations 8[a] and 18[a].

- | | |
|--------------|--|
| [8b] | at least one determination of one or more outputs of at least one processing job based, at least in part, on the aggregating. |
| [18b] | determining one or more outputs of at least one processing job based, at least in part, on the aggregation. |

172. It is my opinion that Foster teaches this limitation.

173. For example, Figure 7-1 shows the result of aggregating the analyzed data from Streams and BigInsights to output visual elements to a user interface or display. Thus, they are the **determination of the outputs of at least one processing job** to “[v]isualize [] real-time and historical insights.” Ex-1004, Fig. 7-1 at 00339. The processing for aggregation and visualization are a **processing job**, at least because it involves processors executing functions on data. As the arrows in Figure 7-1 illustrate, the visualization is based on the aggregating that I just explained above for limitation 8[a]. Foster’s teaching is commensurate with the ’693 patent’s example of “determining one or more outputs of at least one processing job based, at least in part, on the aggregation,” where “a processing mechanism element and/or a service provider may utilize an aggregated output to provide an output to another element and/or another service provider.” Ex-1001, ¶15:41-52. Like the ’693 patent’s example, Foster’s platform outputs the aggregated analytics to another element that processes the data for display.

174. For at least these reasons, the combination of Foster and Williams also teaches limitations [8b] and [18b].

11. Dependent Claims 9 and 19

175. It is my opinion that Foster alone, and Foster combined with Williams renders claims 9 and 19 obvious.

- | |
|--|
| <p>9[a]. The method of claim 1, wherein the at least one static processing mechanism operates, at least in part, in a batch mode with a predetermined batch frequency; and</p> <p>19[a]. The apparatus of claim 11, wherein the at least one static processing mechanism operates, at least in part, in a batch mode with a predetermined batch frequency; and</p> |
|--|

176. Foster alone and the combination of Foster and Williams each teaches claims 1 and 11, as I explained above. *Supra* §§V.A.3-4. It is my opinion that Foster teaches the additional limitation.

177. Foster teaches the **static processing mechanism operates...in a batch mode**. For example, Foster discloses that BigInsights processes batched data from Streams. Ex-1004, §7.1.2 at 00339 (“Streams is used to *batch* and buffer data and, in parallel, *load it into BigInsights* for best performance.”). For example, CDR records are received “periodically as batches of files.” Ex-1004, §7.1.2 at 00340. Batched data is also stored at BigInsights. Ex-1004, §7.1.6 at 00343 (“Sink operator *batches* the data stream into configurable sized chunks for efficient storage in BigInsights.”). It would have been obvious for BigInsights to operate in a batch mode to process such data in batches; indeed, it is a static processing mechanism (as I explained above for limitation [1d]) that is designed to operate in this manner.

178. Foster teaches BigInsight **operates...in a batch mode with a predetermined batch frequency**. For example, Foster explains that BigInsights

uses a “MapReduce framework” and Hadoop file system. Ex-1004, §7.1 at 00338. It was well-known that map-reduce processing was done according to batches on a predetermined schedule. For example, it was known that “[i]n MapReduce environments, many production jobs are run *periodically* on new data.” Ex-1012, §1; *supra* §I.B. This is acknowledged by the ’693 patent, which provides “map-reduce batch jobs” as an example of “static processing.” Ex-1001, 5:4-11, 16:26-34. If a batch is processed periodically, then it is processed at **a predetermined batch frequency** with intervals of a given period.

179. It would have been further obvious because BigInsights continuously accumulates large amounts of Big Data and batch processing at predetermined frequencies was well-known in the art. *See, e.g.*, Ex-1013, 00019 (“*Batch processing* involves the accumulation of transactions until a sufficient number has been assembled to make processing efficient or until other considerations, such as a report cycle, initiate processing. The *processing of batches can be daily, weekly, or monthly*, depending on the volume of transactions and other considerations.”); *supra* §I.B.

180. For at least these reasons, the combination of Foster and Williams also teaches limitations 9[a] and 19[a].

[9b] wherein the at least one dynamic processing mechanism, the at least one processing element, or a combination thereof operate, at least in part, in substantially real-time or at a frequency greater than the predetermined batch frequency.

[19b] wherein the at least one dynamic processing mechanism, the at least one processing element, or a combination thereof operate, at least in part, in substantially real-time or at a frequency greater than the predetermined batch frequency.

181. It is my opinion that Foster teaches this limitation.

182. Foster discloses the **dynamic processing mechanism...operate[s]...in substantially real-time**. For example, Foster discloses that Streams operates on incoming streams for “real-time analysis of data in motion.” Ex-1004, §7.2.1 at 00345, §2.1 at 00057. Foster equates data in motion to “current data” and “real-time data.” Ex-1004, §7.1 at 00338, §2.1 at 00057. Additionally, Foster explains that Streams uses models generated by BigInsights “as a basis for analytics on incoming data” received continuously and updates the models “based on *real-time observations*.” Ex-1004, §7.1.1 at 00339.

183. Foster also teaches that Streams operates **at a frequency greater than the predetermined batch frequency** of BigInsights. For example, Streams operates on data real-time or near real-time, and therefore operates at a frequency corresponding to the incoming rate of data. Streams operates at a greater frequency because the incoming rate of data is continuous, whereas BigInsights

operates on historical or slow moving data according to batch processing (e.g., MapReduce), as I explained above. *Supra* §V.A.11.9[a], §V.A.3.[1d].

184. Foster confirms this by stating that Streams “batch[es]” incoming data for BigInsights (Ex-1004, §7.1.2 at 00339), which is put in “storage” at BigInsights (Ex-1004, §7.1.6 at 00343). Thus, Streams operates at a greater frequency because incoming data items are batched and stored to BigInsights where it is processed later.

185. For at least these reasons, the combination of Foster and Williams also teaches limitations [9b] and [19b].

12. Dependent Claims 10 and 20

186. It is my opinion that Foster alone, and Foster combined with Williams renders claims 10 and 20 obvious.

- 10[a]. The method of claim 1, wherein the at least one static processing mechanism is associated with performing, at least in part, slow moving data analytics; and**
- 20[a] The apparatus of claim 11, wherein the at least one static processing mechanism is associated with performing, at least in part, slow moving data analytics; and**

187. Foster alone and the combination of Foster and Williams each teaches claims 1 and 11, as I explained above. *Supra* §§V.A.3-4. It is my opinion that Foster teaches the additional limitation.

188. Foster teaches **the at least one static processing mechanism is associated with performing...slow moving data analytics**. For example, Foster discloses that BigInsights handles “data-at-rest.” Ex-1004, §7.1 at 00338; *supra* §V.A.3.[1d]. Data-at-rest is “data stored in a database.” Ex-1004, §2.1 at 00057. This data comes from “input streams...ingested into BigInsights for *historical storage* and deep analysis.” Ex-1004, §7.1.5 at 00342. Foster explains that such data stored in databases are “static.” Ex-1004, §2.1 at 00058. The data is slow moving, in contrast to “streaming data or data in motion” (real-time data). Ex-1004, §1.1 at 00025.

189. BigInsights performs **slow moving data analytics** on this slow moving data. For example, BigInsights generates/updates models using “analytics such as data-mining, machine-learning, or statistical modeling.” Ex-1004, §7.1.1 at 00338-339. Thus, BigInsights is **associated with performing...slow moving data analytics**.

190. For at least these reasons, the combination of Foster and Williams also teaches limitations 10[a] and 20[a].

- | |
|---|
| <p>[10b] wherein the at least one dynamic processing mechanism, the at least one processing element, or a combination thereof are associated with performing, at least in part, fast moving data analytics.</p> <p>[20b] wherein the at least one dynamic processing mechanism, the at least one processing element, or a combination thereof are associated with performing, at least in part, fast moving data analytics.</p> |
|---|

191. It is my opinion that Foster teaches this limitation.

192. Foster teaches **the at least one dynamic processing mechanism [and] the at least one processing element...are associated with performing...fast moving data analytics.** For example, Foster discloses that Streams operates on incoming streams for “real-time analysis of data in motion.” Ex-1004, §7.2.1 at 00345, §2.1 at 00057. Foster equates data in motion to “current data” and “real-time data.” Ex-1004, §7.1 at 00338, §2.1 at 00057. The data is fast moving, in contrast to “static data” stored in a database. Ex-1004, §2.1 at 00058.

193. Streams performs fast moving data analytics on this fast moving data. For example, Streams uses models generated by BigInsights “as a basis for *analytics* on incoming data” and updates the models “based on *real-time observations*.” Ex-1004, §7.1.1 at 00339. The continuous stream of data received and prepared at Streams is “also forwarded to downstream analytics on Streams.” Ex-1004, §7.1.2 at 00339. Thus, Streams is **associated with performing...fast moving data analytics.**

194. Additionally, Streams determines a processing element from the incoming stream of real-time data, as I explained above for limitation [1a]. *Supra* §V.A.3.[1a]. The thus determined processing elements are processed at Streams for providing real-time analytics as I just explained. Thus, **the processing element** (e.g., data) is also **associated with performing...fast moving data analytics**

195. For at least these reasons, the combination of Foster and Williams also teaches limitations [10b] and [20b].

VI. EDUCATIONAL BACKGROUND AND QUALIFICATIONS

196. This section contains a summary of my educational background, career history, publications, and other relevant qualifications. My full curriculum vitae is attached as Appendix 1 to this declaration.

197. I am a Professor in the Department of Computer Science at Boston University, and the Chair of Academic Affairs for the Faculty of Computing & Data Sciences. From 2013 to 2018 I also served as Chair of the Department of Computer Science. A brief summary of my background is as follows.

198. I earned an undergraduate degree in Biology from Cornell University, which I received in 1982. I received a master's degree in Computer Science from the University of Buffalo in 1989. I received a Ph.D. in Computer Science from the University of Rochester in 1994. The subject of my Ph.D. thesis was "Performance Prediction and Tuning of Parallel Programs."

199. From 1982 to 1984, I worked as a computer programmer for the State of Colorado. From 1984 to 1994, I was employed at Calspan Corporation, a research and development firm in Buffalo, NY, where I rose to the level of Senior Computer Scientist. My work at Calspan focused on development of experimental software and large-scale simulation software in support of contracts between Calspan and the U.S. Department of Defense.

200. In 1994, I joined the faculty of Boston University, initially as an Assistant Professor of Computer Science. I was promoted to the rank of Associate Professor in 2000 and became a full Professor in 2006. I served as Chair of the Department of Computer Science from 2013 to 2018.

201. I have been pursuing research in the area of computer networking and data science since 1994. I have conducted research in a variety of areas related to the Internet and the World Wide Web, mainly through the application of data mining techniques and statistics. Among other areas, I have studied the efficient design of Web servers and content distribution systems. My research in network systems and data science includes research in the field of database and query systems, which are the backbone of services provided over the Internet and the World Wide Web. I have published a number of research papers on topics related to distributed servers and datamining. My work in these areas is highly cited. From 2007 to 2009, I was the Chair of ACM SIGCOMM (the Special

Interest Group in Computer Communication), the main professional organization for scientists in the field of computer networking.

202. I also spent time in industry working on topics related to network services and databases. I cofounded a startup (Commonwealth Network Technologies) which was acquired by Webmanage, Inc., where I had the title of Chief Scientist. At Webmanage, I designed content distribution systems. Webmanage was subsequently acquired by Network Appliance, Inc., as part of their Netcache division, where I became a Technical Director. At Network Appliance, I helped design systems to distribute Web content and populate Web caches. I also served as Chief Scientist for the data science startup Guavus, developing big data solutions for the telecommunications industry.

203. I am co-author of Internet Measurement: Infrastructure, Traffic, and Applications (Wiley Press, 2006), and I am the author of over two hundred research papers. According to Google Scholar, my work has been cited by over 32,000 other publications. According to Google Scholar, as of May 2018, only 11 other researchers worldwide are more cited in the area of computer systems. I hold nine U.S. Patents derived from my Internet related research. I am a past editor of principal journals in related fields: Computer Communication Review, IEEE/ACM Transactions on Networking, Computer Networks, and IEEE Transactions on Computers.

A. Compensation

204. For my efforts in connection with the preparation of this declaration I have been compensated at my standard rate for this type of consulting activity. My compensation is in no way contingent on the results of these or any other proceedings relating to the above-captioned patent.

B. Materials and Other Information Considered

205. I have considered information from various sources in forming my opinions. I have reviewed and considered each of the exhibits listed in the attached Appendix 2 (Materials Considered in the Preparation of This Declaration) in forming my opinions.

VII. UNDERSTANDING OF THE LAW

206. I am not an attorney. In forming my opinions in this Declaration, I applied the relevant legal principles provided to me by counsel, which are summarized in Appendix 4.

VIII. RESERVATION OF RIGHTS

207. My opinions are based upon the information that I have considered to date. I am unaware of any evidence of secondary considerations with respect to the '693 Patent that would render any of the challenged claims non-obvious. I reserve the right, however, to supplement my opinions in the future to respond to any arguments raised by the owner of the '693 Patent and to take into account new information that becomes available to me.

208. I declare that all statements made herein of my knowledge are true, and that all statements made on information and belief are believed to be true, and that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

Executed on October 18, 2024.

By:

A handwritten signature in black ink, appearing to read "Mark E. Crovella". The signature is written in a cursive, flowing style.

Mark E. Crovella, Ph.D.

Appendix 1: Curriculum Vitae of Mark E. Crovella

MARK E. CROVELLA

Computer Science Department
Boston University
111 Cummington St
Boston, MA 02215
Office: (617) 353-8923
Department: (617) 353-8919

Date of Birth: October 28, 1959
Location: Buffalo, New York, USA
Nationality: USA

Current as of March 8, 2024

EDUCATION

Ph.D. Computer Science, University of Rochester, October 1994.
M.S. University of Rochester, May 1991.
M.S. State University of New York at Buffalo, May 1990.
B.S. Cornell University, May 1982.

PROFESSIONAL EXPERIENCE

Professor Department of Computer Science, Boston University, 2006 to present. **Department Chair** 2013 – 2018; **Associate Professor** 2000 – 2006; **Assistant Professor** 1994 – 2000. On leave 2000 – 2001.

Professor Faculty of Computing & Data Sciences, Boston University, 2022 to present. **Chair of Academic Affairs** 2022 – present; **Founding Faculty Member**, 2020 – 2022; **Data Science Faculty Fellow** 2019 – 2022.

Affiliated Faculty Department of Electrical and Computer Engineering, Graduate Program in Bioinformatics, Center for Information Systems and Engineering, and Division of Systems Engineering, Boston University.

Chief Scientist Guavus, Inc., February 2012 to August 2014.

Visiting Faculty Laboratoire d'Informatique de Paris 6 (LIP6); 2003 – 2004 and 2018 – 2019; INRIA Paris 2018 – 2019; Laboratory of Information, Networking and Communication Sciences (LINCS) Paris 2018 – 2019.

Technical Director Network Appliance, Tewksbury, MA. November 2000 to April 2002.

Chief Architect Webmanage Technologies, Chelmsford, MA. May 2000 to acquisition by Network Appliance in November 2000.

Lead Research Scientist epicRealm, Inc., Richardson, TX. January to May 2001.

Senior Computer Scientist Calspan Corporation, Buffalo, NY. August 1984 to August 1994.

HONORS

Distinguished Alumni Award, 2017, University at Buffalo, Department of Computer Science and Engineering.

Innovator of the Year, 2014, Boston University.

IETF/IRTF Applied Networking Research Prize for “Routing State Distance: A Path-based Metric for Network Analysis,” ACM Internet Measurement Conference (IMC), 2012.

Best Student Paper Award for “Intrusion as (Anti)social Communication: Characterization and Detection,” ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD) 2012.

IEEE Fellow, 2011, “For contributions to the measurement and analysis of networks and distributed systems.”

ACM Fellow, 2010, “For contributions to the measurement and analysis of networks and distributed systems.”

ACM SIGMETRICS Test of Time Award for “Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes,” (with Azer Bestavros). One of the three inaugural Test of Time Awards given by SIGMETRICS in 2010. The award is given for a paper published between 1973 and 1999 that has had the greatest impact over the subsequent period.

CNRS Fellowship, Centre National de la Recherche Scientifique (Department Sciences et Technologies de l’Information et de la Communication), for 2003-2004 academic year as Visiting Associate Professor at Laboratoire d’Informatique de Paris 6 (LIP6).

NSF CAREER Award, 1995.

ARPA Fellowship in High Performance Computing, 1993-1994.

DARPA Parallel Processing Fellowship, 1992-1993.

EDITORIAL BOARDS

Steering Committee, IEEE Transactions on Network Science and Engineering, 2016–2019.

Editor, Computer Communication Review, 2005 – 2008.

Editor, Computer Networks, 2004 – 2005.

Editor, IEEE/ACM Transactions on Networking, 2000 – 2006.

Editor, IEEE Transactions on Computers, 2000 – 2004.

FUNDING

“Large Scale Analysis of Configurations and Management Practices in the Domain Name System,” NSF CISE/CNS NeTS Medium, 10/1/23 – 9/30/26, \$525,000.

“Methods for Active Measurement of the Domain Name System,” NSF CISE/CNS IMR, 10/1/23 - 9/30/26, \$256,463.

Google Research Award, 10/1/20, \$30,000.

“Traveling Back to the Future: Using Network Science to Unravel Evolutionary Conserved RNA Binding Proteins and Complexes Associated with Disease,” BU Hariri Institute Research Incubation Award, 2/9/18, \$35,280.

“Analytic Tools for Evolving Path-based Networks,” NSF CISE/CNS NeTS Small, 8/15/16 – 7/31/19, \$499,951.

“Structural Matrix Completion for Data Mining Applications,” NSF CISE/III Small, 9/1/14 – 8/31/17, \$499,702 with 1 other PI.

“Understanding Communication Strategies for Ad Hoc Networks,” NSF CISE/CNS NeTS Small, 8/1/11 – 7/31/14, \$390,853.

“Information and Software Assurance: A Coordinated Approach to Cyber-Situation Awareness Based on Traffic Anomaly Detection,” DoD ARO, 9/1/2011 - 8/31/2015, \$800,000 with 3 other PIs.

“Securing the Open Softphone,” NSF CISE/CNS TC Large, 7/1/10 – 6/30/15, \$2,992,896, with 8 other PIs. REU Supplement, 4/13/11 – \$73,125.

“New Directions in Network Dimensionality Reduction for Routing and Beyond,” NSF CISE/CNS NeTS Small, 7/1/10 – 6/30/13, \$450,000.

“Wide-Aperture Traffic Analysis for Internet Security,” NSF CISE/CNS TC Medium, 9/1/09 – 8/31/13, \$723,053, with 2 other PIs.

Sponsoring Advisor for Postdoctoral Fellow under the *Computing Innovations Fellowship Program*, NSF CIF, 9/1/09 – 8/31/11, \$267,500.

“Passive Methods for Internet Topology Discovery,” National Geospatial Intelligence Agency, 8/1/09 – 7/31/12, \$360,000.

“Instrumentation and Measurement for GENI,” NSF/BBN, 8/1/08 – 8/31/11, \$80,932

“Whole-Network Anomaly Diagnosis,” Intel, 1/1/05 – 12/31/07, \$158,000.

“Coordinate Systems for the Internet,” NSF CISE/ANI, 8/15/03 – 7/31/06, \$222,035.

“Collaborative Research: Modular Strategies for Global internetwork Monitoring,” with 6 other PIs, NSF CISE/CCR ITR, 9/1/03 – 8/31/08, \$1,994,513.

“Research Support for IP Network Characterization,” Sprint Advanced Technology Laboratories, 9/1/03 – 8/31/04, \$29,592.

“Research Support for IP Network Characterization,” with 3 other PIs, Sprint Advanced Technology Laboratories, 9/1/02 – 8/31/03, \$80,000.

“A Control Theoretic Approach to the Design of Internet Traffic Managers,” with 3 other PIs, NSF CISE/ANIR, 9/1/01 – 8/31/05, \$719,994.

“Diagnosis and Control of Network Variability by Massively Accessed Servers,” with 2 other Principal Investigators, NSF CISE/ANI, 9/1/00 – 8/31/05, \$1,210,768.

“Wide Area Web Measurement,” GTE Laboratories, 9/1/00 – 8/31/01, \$20,000.

“Wide Area Web Measurement,” Xerox PARC, 9/1/99 – 8/31/00, \$15,000.

“COMMONWEALTH: Architecture and Protocols for Scalable WWW Service,” with 2 other Principal Investigators, NSF CISE/CCR Experimental Software Systems, 7/15/97 – 6/30/00, \$590,087.

“Research Support for Network Performance Characterization, Analysis, and Evaluation,” Principal Investigator, Hewlett-Packard Company, 9/1/96 – 9/1/97: \$69,404. Renewed for 9/1/97 – 9/1/98: \$88,154. Renewed for 9/1/98 – 9/1/99: \$29,873.

“Practical Performance Prediction for Parallel Programmers,” Principal Investigator, NSF CISE/CCR CAREER, 9/1/95 – 5/31/98, \$91,066

“Research Infrastructure for Parallel and Distributed Systems: Real-Time, Multimedia, and High-Performance,” with 4 other Principal Investigators, NSF CISE Institutional Infrastructure, \$874,019

“Research Instrumentation for Real-Time, Multimedia and High-Performance Computing in Distributed Systems,” with 4 other Principal Investigators, NSF CISE IGR, \$142,744

“Graduate Assistantships in High Performance Computing,” with 5 other Principal Investigators, US Dept. of Education, 9/1/95 – 8/31/98, \$354,645

PROFESSIONAL AFFILIATIONS

Association for Computing Machinery (Fellow)
 IEEE (Fellow)
 ACM SIGCOMM and SIGMETRICS
 IEEE Computer Society and Communication Society

INVITED CONFERENCE TALKS

Invited Speaker, *TMA Experts Summit*, Paris, June 18, 2019.

Invited Panelist, *The 3rd Future Network Development Conference*, Nanjing, China, May 22, 2019.

Keynote Speaker, *IEEE Workshop on Data Science for Networking (DS4N)*, Boston, MA, December 14, 2017.

Invited Panelist, *CSE 50th Anniversary Celebration*, University at Buffalo, Department of Computer Science and Engineering, September 30, 2017.

- Keynote Speaker, *DATANET 16: Workshop on Data-Driven Networking*, Arizona State University, May 13, 2016.
- Invited Speaker, *Workshop on Mathematics of Data Analysis in Cybersecurity*, ICERM, Brown University, October 24, 2014.
- Invited Speaker, *51st Annual Allerton Conference on Communication, Control, and Computing*, University of Illinois at Urbana-Champaign, October 3, 2013.
- Invited Speaker, *Network Links: Connecting Social, Communication and Biological Network Analysis*, Institute for Mathematics and its Applications (IMA), Minneapolis, MN, February 29, 2012.
- Invited Speaker, *Statistics of Networks Workshop*, Isaac Newton Institute for Mathematical Sciences, Cambridge, UK, June 24-25, 2010.
- Keynote Speaker, *NetEval 2009 Workshop, The First Workshop on Performance Evaluation of Next-Generation Networks*, Boston, MA, April 26, 2009.
- Keynote Speaker, *2006 Euro-NGI Workshop on Visions of Future Generation Networks (EuroView2006)*, Würzburg, Germany, August 1, 2006.
- Invited Speaker, *Maryland Hybrid Networks Center*, Advanced Networks Colloquium Series, University of Maryland, December 16, 2005.
- Keynote Speaker, *Workshop on End-to-End Monitoring Techniques and Services (E2EMON)*, Nice, France, May 15, 2005.
- Invited Speaker, *Workshop on Internet Measurement, Modeling, and Analysis*, Statistical Research Center for Complex Systems (SRCCS), Seoul National University, Seoul, South Korea, January 11, 2005.
- Invited Speaker, *Workshop on Measurement, Modeling and Analysis of the Internet*, Institute for Mathematics and its Applications (IMA), Minneapolis, MN, January 12, 2004.
- Keynote Speaker, *First Workshop on Performance of Computational Systems and Communication*, part of the XXII Congress of the Brazilian Computer Society, Florianopolis, Brazil, July 18, 2002.
- Invited Speaker, Institute for Pure and Applied Mathematics (IPAM) Program on Large Scale Communication Networks, Los Angeles, CA, March 18-22, 2002.
- Invited Speaker, *DIMACS Workshop on Internet and WWW Measurement, Mapping and Modeling*, New Brunswick, NJ, February 13-15, 2002.
- Invited Speaker, *Workshop on Job Scheduling Strategies for Parallel Processing*, Cambridge, MA, June 16, 2001.
- Invited Speaker, *COST 257 Final Conference*, Würzburg, Germany.
- Invited Plenary Speaker, *Performance Tools 2000: The 11th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation*, Schaumburg, Illinois, March 27 - 31, 2000.
- Invited Speaker, *Institute for Mathematics and Its Applications Workshop: "Scaling Phenomena in Communication Networks"*, Minneapolis, MN, October 22-24, 1999.
- Invited Speaker, Boston University Minority Engineers' Society, 2nd Annual Technical Exposition, Boston, MA, April 10, 1999.
- Invited Speaker, Workshop on Intelligent Internet Infrastructure, IBM T.J. Watson Laboratory, Yorktown Heights, NY, March 22, 1999.
- Invited Speaker, Workshop on Architectures, Performance, and Applications in Globally Distributed Web Services, Federal University of Minas Gerais (UFMG), Belo Horizonte, Brazil, March 11 - 12, 1998.
- Invited Speaker, Long-Range Dependence and Self-Similar Network Traffic Session, Winter Simulation Conference, Atlanta, GA, Dec. 7-10, 1997
- Invited Speaker, DIMACS/NSF Workshop on "End-to-End Network Modeling and Simulations," Princeton University, Princeton, NJ, October 23-25, 1997.

Invited Panelist, Hot Topic Session On Traffic Characterization, ITC-15, Washington DC, June 22-27, 1997

Invited Speaker, IFIP Working Group 7.3 Workshop, June 19-20, Orcas Island, WA.

Invited Panelist, "Engineering Implications of Long-Range Dependence in Teletraffic Data," MASCOTS '97, January 1997, Haifa, Israel

Invited Speaker, Applied Probability Session of Informs, the Operations Research Society, May 1997, San Diego, CA.

Invited Speaker, Joint Research Conference on Statistics in Quality, Industry and Technology, May 29-31 1996, NIST, Gaithersburg, MD. Sponsored by the *American Statistical Association*, the *Institute of Mathematical Statistics*, E. I. du Pont, and the *National Institute of Standards and Technology*.

PROFESSIONAL ACTIVITIES

Chair, Scientific Advisory Board of the Max Plank Institute for Software Systems, 2024.

Scientific Advisory Board of the Max Plank Institute for Software Systems, 2016 – 2025.

Member, IETF/IRTF Applied Networking Research Prize Selection Committee, 2014 – 2016.

Executive Steering Committee of the Hariri Institute for Computing and Computational Science & Engineering, 2011 – 2014.

Member of the External Review Visiting Committee for the Department of Computer Science at Amherst College, February 2013.

Organized Workshop: "GPU Programming with CUDA" at Boston University, July 24, 2012.

Member, SIGCOMM Rising Star Award Selection Committee, 2011.

Chair, SIGCOMM Test of Time Award Selection Committee, 2011.

Team Mentor for Conrad Awards, Cyber Security Challenge 2011.

External Judge for FCC's Open Internet Apps Challenge, 2011.

General Co-Chair, CoNEXT 2011.

Internet2 perfSONAR Workshop, Executive Committee, 2010.

Chair, ACM SIGCOMM 2007–2009.

Program Co-Chair, IFIP Networking 2010.

Internet2 Network Research Strategic Planning Team, 2009-present.

Internet2 Network Research Review Committee, <http://www.internet2.edu/networkresearch/nrrc.html>, 2009-present.

Advisory Board of CRAWDAD (Community Resource for Archiving Wireless Data At Dartmouth).

Steering Committee, Passive and Active Measurement Workshop (PAM), 2004-2006.

General Chair, Passive and Active Measurement Workshop (PAM) 2005.

Steering Committee, Internet Measurement Conference, 2003-2007.

ACM SIGCOMM Executive Committee, Award Committee Chair, 2003-2006. b

Program Committee Chair, Internet Measurement Conference 2003.

Co-organizer (with C. Lindemann and M. Reiser), Workshop on Internet Performance Modeling, Dagstuhl Castle, Germany, September 26-29, 1999.

Co-organizer (with five others), BU/NSF Workshop on Internet Measurement, Instrumentation, and Characterization; Boston University, August 30, 1999.

Tutorials Chair, SIGMETRICS '99, Atlanta GA.

Program Committees:

- SIGCOMM: 2002, 2006, 2008, 2012, 2017

- SIGMETRICS: 1999, 2000, 2002, 2007, 2009, 2010, 2011, 2012, 2014, 2015, 2016
- CoNEXT: 2005, 2007, 2010, 2011
- Internet Measurement Conference: 2002, 2004, 2007, 2010, 2012, 2015, 2017, 2018, 2020, 2022
- ICDM: 2018
- The Web Conference: 2000, 2001, 2002, 2019, 2022 (Senior PC Member)
- KDD: 2017, 2023
- Intelligent Systems for Molecular Biology (ISMB): 2015, 2017
- First IMC Workshop on Internet Visualization: 2012
- Performance: 2011
- Passive and Active Measurement Workshop (PAM): 2009, 2010
- Infocom: 2009 (TPC Area Chair), 2010 (TPC Area Chair)
- IFIP Networking: 2009
- NetEval: 2009
- HotMetrics: 2008
- E2EMON: 2007
- First Workshop on Experimental Computer Science: 2007
- Workshop on Mining Network Data (MineNet-06): 2006
- Workshop on Large Scale Network Inference: 2005
- 6th Global Internet Symposium: 2001
- 6th International Workshop on Web Caching and Content Distribution: 2001
- Workshop on Caching, Coherence and Consistency: 2001
- Workshop on Network-Related Data Management: 2001
- ITC Specialist Seminar on IP Traffic Management and Modeling: 2000
- SPIE Performance and Control of Network Systems Conference: 1998, 1999
- 18th International Conference on Distributed Computing Systems (ICDCS-98): 1998
- Third IEEE Workshop on Services in Distributed and Networked Environments (SDNE '96): 1996

COURSE TEXTS AND MATERIALS

- “Linear Algebra, Geometry, and Computation,” Electronic Text, one-semester course on Linear Algebra from a Computer Science standpoint. 2015-2022. Available at <http://mcrovella.github.io/CS132-Geometric-Algorithms>. Includes contributions from Wayne Snyder.
- “Computational Tools for Data Science,” Electronic Text, one-semester course on applied methods in machine learning using Python. 2021-2022. Available at <http://mcrovella.github.io/CS506-Computational-Tools-for-Data-Science>. Includes contributions from Evimaria Terzi and George Kollios.

TUTORIALS AND SHORT COURSES

- “Data Mining for Measurement Data,” presented at the 5th PhD School on Traffic Monitoring and Analysis, associated with the 7th International Workshop on Traffic Monitoring and Analysis, Barcelona, April 21-22, 2015. Available at <https://github.com/mcrovella/mining-low-dim-network-data>.
- “Traffic Modeling: Methods and Results for Single Links and Whole Networks,” presented at SIGCOMM 2004, Portland OR.
- “Traffic Modeling,” short course presented at Aalborg University, February 2, 2004.
- “Performance Characteristics of World Wide Web Information Systems,” presented at SIGMETRICS '97, Seattle WA, June 1997; SIGMETRICS '98, Madison WI, June 1998; and Performance Tools '98, Palma de Mallorca, Spain, September 1998.

CONSULTING

Member of the Technical Advisory Board, Guavus, Inc. 2006 to 2017.

Consultant, Fujitsu Laboratories of America, Sunnyvale, CA. 2002 to 2004.

Consultant and Expert Witness, in litigation cases involving software patents, software trade secrets, and intellectual property. Ongoing.

Member of the Technical Advisory Board, epicRealm, Inc., Richardson, TX. March 2000 to May 2001.

Consultant and Member of the Technical Advisory Board, WebManage Technologies, Inc., Chelmsford MA. July 1999 to November 2000.

Founder and Principal Technical Officer, Commonwealth Network Technologies, Inc., Boston MA. Incorporated April 27, 1998; acquired by WebManage Technologies, October 12, 1999.

Consultant and Member of the Technical Advisory Board, VitalSigns Software Inc., Santa Clara, CA; acquired by International Network Services, Inc.; now part of Lucent. November 1996 to October 1999.

PATENTS

Methods for Storing and Reading Digital Data on a Set of DNA Strands, US Patent # 10,027,347, issued July 17, 2018.

Method and Apparatus for Whole-Network Anomaly Diagnosis and Method to Detect and Classify Network Anomalies Using Traffic Feature Distributions, US Patent # 8,869,276, issued October 21, 2014.

Compressed Set Representation for Sets as Measures in OLAP Cubes, US Patent # 8,533,167, issued September 10, 2013.

Method and Apparatus for Dynamic Resource Discovery and Information Distribution in a Data Network, US Patent # 7,747,741, issued June 29, 2010.

System for Creating and Distributing Prioritized List of Computer Nodes Selected as Participants in a Distribution Job, US Patent # 7,346,682, issued March 18, 2008.

Method and Apparatus for Election of Group Leaders in a Distributed Network, US Patent # 6,993,587, issued January 31, 2006.

Method and Apparatus for Scalable Distribution of Information in a Distributed Network, US Patent # 6,748,447, issued June 8, 2004.

Method and Apparatus for Reliable and Scalable Distribution of Data Files in Distributed Networks, US Patent # 6,718,361, issued April 6, 2004. European Patent # EP1305924 issued March 31, 2010.

Distributed Routing, US Patent # 6,370,584, issued April 9, 2002.

Method and Apparatus for Assigning Tasks in a Distributed Server System, US Patent # 6,223,205, issued April 24, 2001.

DOCTORAL STUDENTS

Ahmed Youssef, Ph.D. June 2023. “Computational Methods to Uncover Cell State Proteomes and Profile Protein Interaction Dynamics.” Co-advised with Andrew Emili.

Bashir Rastegarpanah, Ph.D. August 2021. “Tools for Responsible Decision-Marking in Machine Learning.”

Larissa Spinelli, Ph.D. June 2019. “Empirical Studies of Factors Affecting Opinion Dynamics.”

Giovanni Comarela, Ph.D. July 2017. “On the Dynamics of Interdomain Routing in the Internet.”

Natali Ruchansky, Ph.D. August 2016. “Matrix Completion with Structure.” Co-advised with Evimaria Terzi.

Andrej Cvetkovski, Ph.D. May 2013. “Graph Embeddings for Low-Stretch Greedy Routing.”

Gonca Gürsun, Ph.D. March 2013. “Inferring Hidden Features in the Internet.”

Vijay Erramilli, Ph.D. December 2008. “Forwarding in Mobile Opportunistic Networks.”

Nahur Fonseca, Ph.D. December 2008. “Stochastic Modeling Applied to Detection Problems in Network Protocols and Traffic.”

Anukool Lakhina, Ph.D. August 2006. “Network-Wide Traffic Analysis: Methods and Applications.”

Jun Liu, Ph.D. August 2002. “Characterizing Network Elements and Paths Using Packet Loss Behavior.”

Paul Barford, Ph.D. June 2000. “Modeling, Measurement and Performance of World Wide Web Transactions.”

Robert L. Carter, Ph.D. January 1998. “Performance Measurement and Prediction in Packet-Switched Networks: Techniques and Applications.”

OTHER DOCTORAL STUDENTS ADVISED

Mengfei Cao, Ph.D. August 2016, “New Graph Metrics Improve Network-based Protein Function Prediction” Tufts University, Member of the Committee.

Bimal Viswanath, Ph.D. April 2016, “Towards Trustworthy Social Computing Systems.” Universität des Saarlandes, Member of the Committee.

Yonjun Liao, Ph.D., January 2013, “Learning to Predict End-to-End Network Performance.” Université de Liège, Belgium, Member of the Jury.

Diana Zeaiter Jomblatt, Ph.D., December 2012, “Prediction of User Dissatisfaction with Internet Application Performance at End-Hosts.” Université Pierre et Marie Curie, Rapporteur.

Chrisil Arackaparambil, Ph.D., July 2011, “Anomaly Detection in Network Streams Through a Distributional Lens.” Dartmouth College, Member of the Examining Committee.

Italo Cunha, Ph.D., July 2011, “Tracking Internet Routes and Path Reachability for Network Tomography.” Université Pierre et Marie Curie (U. Paris VI), Rapporteur.

Hung Nguyen, Ph.D., February 2008, “User-level Internet Tomography and Overlay Routing.” Ecole Polytechnique Fédérale de Lausanne (EPFL), Rapporteur.

Benoit Donnet, Ph.D., September 2006, “Algorithms for Large-Scale Topology Discovery.” Université Pierre et Marie Curie (U. Paris VI), Président du Jury.

David Chua, Ph.D. (Mathematics), May 2006, “Statistical Analysis for Whole Networks.” Second Reader.

Stilian Stoev, Ph.D. (Mathematics) May 2005.

Abhishek Kumar, Ph.D., October 2005, “Network Data Streaming: Algorithms for Network Monitoring and Coordination.” Georgia Tech, Committee Member.

Marjan Bozinovski, Ph.D., May 2004, “Fault-Tolerant Platforms for IP-Based Session Control Systems.” Aalborg University, Committee Member.

Artur Ziviani, Ph.D. December 2003, “Qualité de Service et Localisation d’Hotes.” University of Paris VI, Examineur.

Liang Guo, Ph.D. April 2003, “Size-Aware Scheduling of TCP Flows.” Second Reader.

Alberto Medina, Ph.D. March 2003, “Practical Estimation of Internet Traffic Demands.” Third Reader.

Vladas Pipiras, Ph.D. (Mathematics) April 2002, “Symmetric Stable Self-Similar Processes.” Third Reader.

Khaled A. Harfoush, Ph.D. November 2001, “A Framework for End-to-End Characterization of Metric-Induced Network Topologies.” Committee Member.

Santiago M. Pericas-Geertsen, Ph.D. November 2001, “XML-Fluent Mobile Agents.” Committee Member.

- Alia K. Atlas**, Ph.D. August 1998, “Statistical Rate Monotonic Scheduling.” Second Reader.
- Ghulam Bhatti**, Ph.D. January 1998, “Parallel and Distributed Discrete Event Simulation: An Event-Reservation Approach.” Second Reader.
- Sulaiman Mirdad**, Ph.D. June 1997, “WebWave: Globally Load-Balanced, Fully-Distributed Caching”. Second Reader.
- Gi Tae Kim**, Ph.D. June 1997, “Framework for Adaptive Forward Erasure Recovery (AFER) for Fault-Tolerant Real-Time Communications”. Second Reader.
- Carlos Cunha**, Ph.D. June 1997, “Trace Analysis and its Application to Performance Enhancements of Distributed Information Systems”. Second Reader.
- Susan Nagy**, Ph.D. April 1997, “Admission Control and Scheduling Strategies for Real-Time Database Systems”. Second Reader.
- Vadim Teverovksy**, Ph.D. February 1997 (Mathematics), “Detection and Estimation of Long-Range Dependence.” Third Reader.
- Euthimios Panagos**, Ph.D. May 1995, “Client-Based Logging: A New Paradigm for Distributed Transaction Management”. Third Reader.

MASTERS STUDENTS

- Megan Hacker**, M.S. May 2012, “Methods for Pairing Mobile Devices Through Shared Sensing”
- Lauren Walters**, MSc. May 2004, “A Web Browsing Workload Model for Simulation,” University of Cape Town, South Africa, external examiner.
- Alan Charbonneau**, M.A. May 2003, “Simulation of an SJF Proxy”
- Iana Vitkova**, M.A. May 2003, “Simulation of an SJF Proxy”
- Robert Frangioso**, M.A. Sept 2002, “Connection Scheduling in Web Servers”
- Xiaohui Zhang**, M.S. May 1999, “Cachability of Web Objects”
- Scott Harrison**, M.S. May 1998, “DPR Monitor: A Web-based DPR Network Monitoring Tool”
- Stanley Gambarin**, M.S. May 1997, “Implementing a Single-Process Non-Multithreaded World Wide Web Server”
- Kostas Magoutis**, M.S. May 1996, “Economies of Scale In Partitioned Disk Caches with Self-Similar References”
- Fred Bierhaus**, M.S. May 1996, “Improving Interactive Traffic Through the EQL Driver”

BOOKS

- M. Crovella and B. Krishnamurthy, “Internet Measurement: Infrastructure, Traffic, and Applications,” John Wiley and Sons, 2006, 512 pp.

BOOKS EDITED

- M. Crovella and L. M. Feeney and D. Rubenstein (editors), “Proceedings of the 9th International IFIP TC 6 Networking Conference,” Springer-Verlag, 410pp, 2010.

JOURNAL ARTICLES

- A. Youssef, I. Paul, M. Crovella, and A. Emili, “DESP: Demixing Cell State Profiles from Dynamic Bulk Molecular Measurements,” *Cell Reports Methods*, 2024.
- L. Salamatian, S. Anderson, J. Matthews, P. Barford, W. Willinger, and M. Crovella, “A Manifold View of Connectivity in the Private Backbone Networks of Hyperscalers,” *Communications of the ACM*, August 2023.

- N. Kewalramani, A. Emili, M. Crovella, “State-of-the-Art Computational Methods to Predict Protein-Protein Interactions with High Accuracy and Coverage,” *Proteomics*, 2023.
- A. Youssef, F. Bian, P. Havugimana, M. Crovella, and A. Emili, “Dynamic Remodeling of *Escherichia coli* Interactome in Response to Environmental Perturbations,” *Proteomics*, 2023.
- L. Salamatian, S. Anderson, J. Mathews, P. Barford, W. Willinger, and M. Crovella, “A Manifold View of Connectivity in the Private Backbone Networks of Hyperscalers,” *Communications of the ACM*, Research Highlights, 6(7), July 2023.
- J. Jung, M. Guo, M. Crovella, J.G. McDaniel, and K.M. Warkentin, “Frog Embryos Use Multiple Levels of Temporal Pattern in Risk Assessment for Vibration-Cued Hatching,” *Animal Cognition*, 2022.
- J. N. Law, K. Akers, N. Tasnina, C. M. Della-Santina, S. Deutsch, M. Kshirsagar, J. Klein-Seetharaman, M. Crovella, P. Rajagopalan, S. Kasif, and T. M. Murali, “Interpretable Network Propagation with Application to Expanding the Repertoire of Human Proteins that Interact with SARS-CoV-2,” *GigaScience*, 10(12), December 2021.
- D. Lancour, J. Dupuis, R. Mayeux, J. Haines, M. Pericak-Vance, G. Schellenberg, M. Crovella, L. Farrer, and S. Kasif, “Analysis of Brain Region-Specific Co-Expression Networks Reveals Clustering of Established and Novel Genes Associated with Alzheimer Disease,” *Alzheimer’s Research & Therapy*, 2020.
- J. Fan, X. C. Li, M. Crovella, and M. D. M. Leiserson, “Matrix (Factorization) Reloaded: Flexible Methods for Imputing Genetic Interactions with Cross-Species and Side Information,” *OUP Bioinformatics*, 2020.
- A. Ramirez, S. Dankel, B. Rastegarpanah, W. Cai, R. Xue, M. Crovella, Y.-H. Tseng, R. Kahn, and S. Kasif, “Single-Cell Transcriptional Networks in Differentiating Preadipocytes Suggest Drivers Associated with Tissue Heterogeneity,” *Nature Communications*, 2020.
- J. Fan, A. Cannistra, I. Fried, T. Lim, T. Schaffner, M. Crovella, B. Hescott, and M.D.M. Leiserson, “Functional Protein Representations from Biological Networks Enable Diverse Cross-Species Inference,” *Nucleic Acids Research*, 2019.
- D. Lancour, A. Naj, R. Mayeux, J. L. Haines, M. A. Pericak-Vance, G. C. Schellenberg, M. Crovella, L. A. Farrer, and S. Kasif, “One for All and All for One: Improving Replication of Genetic Studies through Network Diffusion,” *PLoS Genetics*, Vol 14, April 2018.
- A. Cvetkovski and M. Crovella, “Low-stress Data Embedding in the Hyperbolic Plane Using Multidimensional Scaling,” *Applied Mathematics and Information Sciences*, Vol 11, Number 1, January 2017.
- A. Cvetkovski and M. Crovella, “Multidimensional Scaling in the Poincaré Disk,” *Applied Mathematics and Information Sciences*, Vol 10, Number 1, January 2016.
- M. Cao, H. Zhang, J. Park, N. Daniels, M. Crovella, L. Cowen, and B. Hescott, “Going the Distance for Protein Function Prediction: A New Distance Metric for Protein Interaction Networks,” *PLoS One*, Vol 8, Number 10, October 2013.
- A. Cvetkovski and M. Crovella, “On the Choice of a Spanning Tree for Greedy Embedding of Network Graphs,” *Networking Science*, February 2013.
- B. Donnet, P. Raoult, T. Friedman, and M. Crovella, “Deployment of an Algorithm for Large-Scale Topology Discovery,” *IEEE Journal on Selected Areas in Communications, Special Issue on Sampling the Internet*, Vol 24, Number 12, December 2006.
- D. Chua, E. Kolaczyk, and M. Crovella, “Network Kriging,” *IEEE Journal on Selected Areas in Communications, Special Issue on Sampling the Internet*, 2006.
- B. Gueye, A. Ziviani, M. Crovella, and S. Fdida, “Constraint-Based Geolocation of Internet Hosts,” *IEEE/ACM Transactions on Networking*, Vol 14, Number 6, December 2006.

- R. Fonseca, V. Almeida, and M. E. Crovella, "Locality in a Web of Streams," *Communications of the ACM*, 48(1), pp. 82–88, January 2005.
- M. Fayed, P. Krapivsky, J. Byers, M. Crovella, D. Finkel, and S. Redner, "On the Emergence of Highly Variable Distributions in the Autonomous System Topology," *Computer Communication Review*, 33(2), pp. 41–49, July 2003.
- A. Lakhina, J. W. Byers, M. E. Crovella, and I. Matta, "On the Geographic Location of Internet Resources," *IEEE Journal on Selected Areas in Communications, Special Issue on Internet and WWW Measurement, Mapping, and Modeling*, 2003.
- P. Barford and M. E. Crovella, "Critical Path Analysis of TCP Transactions," *IEEE/ACM Transactions on Networking*, 9 (3), pp. 238–248.
- M. E. Crovella, C. Lindemann, and M. Reiser, "Internet Performance Modeling: the state of the art at the turn of the century," *Performance Evaluation*, 42 (2–3), pp. 91–108, September 2000.
- R. L. Carter and M. E. Crovella, "On the Network Impact of Dynamic Server Selection," *Computer Networks*, 31 (23–24), pp. 2529–2558.
- M. Harchol-Balter, M. E. Crovella and C. D. Murta, "On Choosing a Task Assignment Policy for a Distributed Server System," *Journal of Parallel and Distributed Computing*, Special Issue on Software Support for Distributed Computing, September 1999.
- P. Barford and M. E. Crovella, "Measuring Web Performance in the Wide Area," *Performance Evaluation Review*, Special Issue on Network Traffic Measurement and Workload Characterization, August 1999.
- P. Barford, A. Bestavros, A. Bradley, and M. E. Crovella, "Changes in Web Client Access Patterns: Characteristics and Caching Implications," *World Wide Web*, Special Issue on World Wide Web Characterization and Performance Evaluation, 1999.
- M. E. Crovella and M. S. Taqqu, "Estimating the Heavy Tail Index from Scaling Properties," *Methodology and Computing In Applied Probability*, 1(1), 1999.
- M. E. Crovella and A. Bestavros, "Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes," *IEEE/ACM Transactions on Networking*, 5(6):835–846, December 1997.
- R. L. Carter and M. E. Crovella, "Measuring Bottleneck Link Speed in Packet-Switched Networks," *Performance Evaluation*, 27&28:297–318, October 1996.
- L. A. Cowl, M. E. Crovella, T. J. LeBlanc, and M. L. Scott, "The Advantages of Multiple Parallelizations in Combinatorial Search," *Journal of Parallel and Distributed Computing*, special issue on Data-Parallel Algorithms and Programming, 21(1):110–123, April 1994.

BOOK CHAPTERS

- M. E. Crovella, M. S. Taqqu, and A. Bestavros, "Heavy-Tailed Probability Distributions in the World Wide Web," Chapter 1 of *A Practical Guide To Heavy Tails*, pages 3–25, Chapman & Hall, New York, 1998.
- M. E. Crovella and L. Lipsky, "Simulations with Heavy-Tailed Workloads," Chapter 3 of *Self-Similar Network Traffic and Performance Evaluation*, Wiley/Wiley-Interscience, 1999.
- K. Park, G. Kim and M. Crovella, "The Protocol Stack and its Modulation Effect on Self-Similar Traffic," Chapter 16 of *Self-Similar Network Traffic and Performance Evaluation*, Wiley/Wiley-Interscience, 1999.
- M. E. Crovella, "Performance Characteristics of the World Wide Web," in *White Book on Performance Evaluation*, Günter Haring, Christoph Lindemann, and Martin Reiser, editors, Springer-Verlag, 1999.

INVITED PAPERS

- M. E. Crovella, "Performance Evaluation with Heavy Tailed Distributions," in *Lecture Notes in Computer Science 1786*, Proceedings of the 11th International Conference on Modelling Tools and Techniques for Computer and Communication System Performance Evaluation (TOOLS '2000), pp. 1-9, March 2000. Revised version appeared in *Job Scheduling Strategies for Parallel Processing : 7th International Workshop*, pp. 1-10, June 16, 2001.

REFEREED CONFERENCE PAPERS

- A. Anderson, A. S. Mondal, P. Barford, M. Crovella, and J. Sommers, "An Elemental Decomposition of DNS Name-to-IP Graphs," *Proceedings of INFOCOM*, Vancouver, Canada, May 2024.
- G. Franco, M. Crovella, and G. Comarela, "Dependence and Model Selection in LLP: The Problem of Variants," *Proceedings of KDD*, Los Angeles, CA, August 2023.
- K. Quinn, E. Terzi, and M. Crovella, "Characterizing Covid Waves via Spatio-Temporal Decomposition," *Proceedings of KDD*, Washington DC, August 2022.
- L. Salamatian, S. Anderson, J. Matthews, P. Barford, W. Willinger, and M. Crovella, "Curvature-based Analysis of Network Connectivity in Private Backbone Infrastructures," *Proceedings of ACM SIGMETRICS / IFIP Performance*, Mumbai, India, June 2022.
- B. Rastegarpanah, K. P. Gummadi, and M. Crovella, "Auditing Black-Box Prediction Models for Data Minimization Compliance," *Proceedings of NeurIPS*, Online, December 2021. **Selected as a Spotlight Presentation (3% acceptance rate).**
- L. Amichi, A. Carneiro Viana, M. Crovella, and A. A. F. Loureiro, "From Motion Purpose to Perceptive Spatial Mobility Prediction," *Proceedings of ACM SIGSPATIAL*, Beijing, China, November 2021.
- G. Grassi, C. Barakat, M. Crovella, and R. Teixeira, "Leveraging Website Popularity Differences to Identify Performance Anomalies," *Proceedings of Infocom*, Online, May 2021.
- L. Amichi, A. Carneiro Viana, M. Crovella, and A. A. F. Loureiro, "Understanding Individuals' Proclivity for Novelty Seeking," *Proceedings of ACM SIGSPATIAL*, Online, November 2020.
- J. Fan, X. C. Li, M. Crovella, and M. D. M. Leiserson, "Matrix (Factorization) Reloaded: Flexible Methods for Imputing Genetic Interactions with Cross-Species and Side Information," *Proceedings of the 19th European Conference on Computational Biology*, Online, August 2020.
- B. Rastegarpanah, M. Crovella, and K. P. Gummadi, "Fair Inputs and Fair Outputs: the Incompatibility of Fairness in Privacy and Accuracy," *Proceedings of the Workshop on Fairness in User Modeling, Adaptation, and Personalization (FairUMAP)*, Online, July 2020.
- L. Spinelli and M. Crovella, "How YouTube Leads Privacy-Seeking Users Away from Reliable Information," *Proceedings of the Workshop on Fairness in User Modeling, Adaptation, and Personalization (FairUMAP)*, Online, July 2020.
- J. Law, M. Kshirsagar, M. Crovella, P. Rajagopalan, J. Klein-Seetharaman, S. Kasif and T. M. Murali, "A Network-Based Label Propagation Framework to Reposition Drugs for COVID-19 (Poster)," *Proceedings of RECOMB*, Online, June 2020.
- L. Amichi, A. Viana, M. Crovella, and A. A. F. Loureiro, "Mobility profiling: Identifying Scouters in the Crowd (Poster)," *Proceedings of the CoNEXT Student Workshop*, Orlando, FL, December 2019.
- B. Rastegarpanah, K. P. Gummadi, and M. Crovella, "Fighting Fire with Fire: Using Antidote Data to Improve Polarization and Fairness of Recommender Systems," *Proceedings of WSDM 2019*, Melbourne, Australia, February 2019.

- G. Comarela, R. Durairajan, P. Barford, D. Christenson, and M. Crovella, "Assessing Candidate Preference through Web Browsing History," *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, London, UK, August 2018. Press coverage: BU Today: <http://www.bu.edu/today/2018/better-election-predictions>.
- L. Spinelli and M. Crovella, "Unravelling the Dynamics of Online Ratings," *Proceedings of the 4th International Symposium on Social Media Mining and Analysis (SMMA 2018)*, Exeter, UK, June 2018.
- M.D.M. Leiserson, J. Fan, A. Cannistra, I. Fried, T. Lim, T. Schaffner, M. Crovella, and B. Hescott, "A Multi-Species Functional Embedding Integrating Sequence and Network Structure," *Proceedings of RECOMB*, Paris, France, April 2018.
- B. Rastegarpanah, K. Gummadi, and M. Crovella, "Exploring Explanations for Matrix Factorization Recommender Systems" (Position Paper), *FATREC Workshop on Responsible Recommendation*, Como Italy, August 2017.
- L. Spinelli and M. Crovella, "Closed-Loop Opinion Formation", *Proceedings of the 9th International ACM Web Science Conference (WebSci)*, Troy, NY, June 2017.
- N. Ruchansky, E. Terzi, and M. Crovella, "Targeted Matrix Completion," *Proceedings of the SIAM International Conference on Data Mining (SDM)*, Houston, TX, April 2017.
- C. Mavroforakis, D. Erdős, M. Crovella, and E. Terzi, "Active Positive-Definite Matrix Completion," *Proceedings of the SIAM International Conference on Data Mining (SDM)*, Houston, TX, April 2017.
- G. Comarela, E. Terzi, and M. Crovella, "Detecting Unusually-Routed ASes: Methods and Applications," *Proceedings of the ACM Internet Measurement Conference (IMC)*, Santa Monica, CA, November 2016.
- N. Ruchansky, M. Crovella, and E. Terzi, "Matrix Completion with Queries," *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, Sydney, Australia, August 2015.
- G. Comarela and M. Crovella, "Identifying and Analyzing High Impact Routing Events with PathMiner," *Proceedings of the ACM Internet Measurement Conference (IMC)*, Vancouver, BC, November 2014.
- B. Viswanath, M. A. Bashir, M. Crovella, S. Guha, K. Gummadi, B. Krishnamurthy, and A. Mislove, "Towards Detecting Anomalous User Behavior in Online Social Networks," *Proceedings of the 23rd USENIX Security Symposium*, San Diego, CA, August 2014.
- Y. Zhang, T. Lappas, M. Crovella, and E. Kolaczyk, "Online ratings: convergence towards a positive perspective?," *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Florence, Italy, May 2014.
- G. Comarela, G. Gürsun, and M. Crovella, "Studying Interdomain Routing over Long Timescales," *Proceedings of the ACM Internet Measurement Conference (IMC)*, Barcelona, Spain, October 2013.
- L. Spinelli, M. Crovella and B. Eriksson, "AliasCluster: A Lightweight Approach to Interface Disambiguation," *Proceedings of the Global Internet Symposium*, Turin, Italy, April 2013.
- B. Eriksson and M. Crovella, "Understanding Geolocation Accuracy using Network Geometry," in *Proceedings of the Infocom 2013 Miniconference*, Turin, Italy, April 2013.
- J. M. Agosta, J. Chandrasekar, M. Crovella, N. Taft, and D. Ting, "Mixture Models of Endhost Network Traffic," in *Proceedings of the Infocom 2013 Miniconference*, Turin, Italy, April 2013.
- G. Gürsun, N. Ruchansky, E. Terzi, and M. Crovella, "Routing State Distance: A Path-based Metric for Network Analysis," in *Proceedings of the ACM Internet Measurement Conference (IMC)*, Boston, MA, November 2012. **Winner of a 2013 IETF/IRTF Applied Networking Research Prize.**
- G. Gürsun and M. Crovella, "On Traffic Matrix Completion in the Internet," in *Proceedings of the ACM Internet Measurement Conference (IMC)*, Boston, MA, November 2012.

- T. Lappas, M. Crovella, and E. Terzi, "Selecting a Characteristic Set of Reviews," in *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD) 2012*, Beijing, China, August 2012.
- Q. Ding, N. Katenka, P. Barford, E. Kolaczyk, and M. Crovella, "Intrusion as (Anti)social Communication: Characterization and Detection," in *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD) 2012*, Beijing, China, August 2012. **Best Student Paper Award.**
- G. Gürsun, N. Ruchansky, E. Terzi, and M. Crovella, "Inferring Visibility: Who's (not) Talking to Whom?," in *Proceedings of SIGCOMM 2012*, Helsinki, Finland, August 2012.
- B. Eriksson and M. Crovella, "Estimating Intrinsic Dimension Via Clustering," in *IEEE Statistical Signal Processing Workshop (SSP)*, Ann Arbor, MI, August 2012.
- C. Thomas, J. Sommers, P. Barford, D. Kim, A. Das, R. Segebre, M. Crovella, "A Passive Measurement System for Network Testbeds," in *Tridentcom 2012*, Thessaloniki, Greece, June 2012.
- G. Comarela, M. Crovella, V. Almeida and F. Benevenuto, "Understanding Factors that Affect Response Rates in Twitter," in *ACM Hypertext 2012*, Milwaukee, WI, June 2012.
- A. Cvetkovski and M. Crovella, "Low-Stretch Greedy Embedding Heuristics," in *Fourth International Workshop on Network Science for Communication Networks (NetSciCom 2012)*, Orlando, FL, March 2012.
- V. Manfredi, M. Crovella, and J. Kurose, "Understanding Stateful vs Stateless Communication Strategies for Ad hoc Networks," in *Proceedings of Mobicom 2011*, Las Vegas, NV, September 2011.
- G. Gürsun, M. Crovella, and I. Matta, "Describing and Forecasting Video Access Patterns," in *Proceedings of the Infocom 2011 Miniconference*, Shanghai, China, April 2011.
- V. Bharti, P. Kankar, L. Setia, G. Gürsun, and M. Crovella, "Inferring Invisible Traffic," in *Proceedings of CoNEXT 2010*, Philadelphia, PA, December 2010.
- J. Sommers, P. Barford and M. Crovella, "Router Primitives for Programmable Active Measurement," in *Workshop on Programmable Routers for Extensible Services of Tomorrow (PRESTO '09)*, Barcelona, Spain, August, 2009.
- A. Cvetkovski and M. Crovella, "Hyperbolic Embedding and Routing for Dynamic Graphs," in *Proceedings of Infocom 2009*, Rio de Janeiro, Brazil, April 2009.
- V. Erramilli and M. Crovella, "Forwarding in Opportunistic Networks with Resource Constraints," in *Proceedings of the Fourth ACM Workshop on Challenged Networks (CHANTS 08)*, San Francisco, CA, September, 2008.
- N. Fonseca, M. Crovella, and K. Salamatian, "Long Range Mutual Information," in *Proceedings of the First Workshop on Hot Topics in Measurement and Modeling of Computer Systems (Hotmetrics '08)*, Annapolis, MD, June, 2008.
- V. Erramilli, A. Chaintreau, M. Crovella, and C. Diot, "Delegation Forwarding," in *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (ACM MobiHoc)*, Hong Kong SAR, China, May 2008.
- P. Chhabra, C. Scott, E. D. Kolaczyk, and M. Crovella, "Distributed Spatial Anomaly Detection," in *Proceedings of Infocom 2008*, Phoenix, AZ, April 2008.
- V. Erramilli, A. Chaintreau, M. Crovella, and C. Diot, "Diversity of Forwarding Paths in Pocket Switched Networks," in *Proceedings of the ACM SIGCOMM Internet Measurement Conference*, San Diego, October, 2007.
- B. Eriksson, P. Barford, R. Nowak, and M. Crovella, "Learning Network Structure from Passive Measurements," in *Proceedings of the ACM SIGCOMM Internet Measurement Conference*, San Diego, October, 2007.
- V. Erramilli, M. Crovella, and N. Taft, "An Independent-Connection Model for Traffic Matrices," in *Proceedings of the ACM SIGCOMM Internet Measurement Conference*, Rio de Janeiro, October, 2006.

- X. Li, F. Bian, M. Crovella, C. Diot, R. Govindan, G. Iannaccone, and A. Lakhina, "Detection and Identification of Network Anomalies Using Sketch Subspaces," in *Proceedings of the ACM SIGCOMM Internet Measurement Conference*, Rio de Janeiro, October, 2006.
- A. Lakhina, M. Crovella, and C. Diot, "Detecting Distributed Attacks using Network-Wide Flow Traffic," in *Proceedings of FloCon 2005 Analysis Workshop*, New Orleans, LA, September, 2005.
- A. Lakhina, M. Crovella, and C. Diot, "Mining Anomalies Using Traffic Feature Distributions," in *Proceedings of ACM SIGCOMM*, Philadelphia, PA, August 2005.
- D. B. Chua, E. D. Kolaczyk, and M. Crovella, "A Statistical Framework for Efficient Monitoring of End-to-End Network Properties," (Poster) in *Proceedings of ACM SIGMETRICS*, Banff, Canada, June 2005.
- B. Donnet, P. Raoult, T. Friedman, and M. Crovella, "Efficient Algorithms for Large-Scale Topology Discovery," in *Proceedings of ACM SIGMETRICS*, Banff, Canada, June 2005.
- A. Soule, A. Lakhina, N. Taft, K. Papagiannaki, K. Salamatian, A. Nucci, M. Crovella, and C. Diot, "Traffic Matrices: Balancing Measurements, Inference and Modeling," in *Proceedings of ACM SIGMETRICS*, Banff, Canada, June 2005.
- B. Donnet, T. Friedman, and M. Crovella, "Improved Algorithms for Network Topology Discovery," in *Lecture Notes in Computer Science, Proceedings of the 2005 Passive and Active Measurement Workshop*, Boston, MA, March, 2005.
- D. B. Chua, E. D. Kolaczyk, and M. Crovella, "Efficient Monitoring of End-to-End Network Properties," in *Proceedings of Infocom 2005*, Miami, FL, March 2005.
- N. Fonseca and M. Crovella, "Bayesian Packet Loss Detection for TCP," in *Proceedings of Infocom 2005*, Miami, FL, March 2005.
- B. Gueye, A. Ziviani, M. Crovella, and S. Fdida, "Constraint-Based Geolocation of Internet Hosts," in *Proceedings of the ACM SIGCOMM Internet Measurement Conference*, Taormina, Italy, October 2004.
- A. Lakhina, M. Crovella, and C. Diot, "Characterization of Network-Wide Anomalies in Traffic Flows," in *Proceedings of the ACM SIGCOMM Internet Measurement Conference*, Taormina, Italy, October 2004.
- A. Lakhina, M. Crovella, and C. Diot, "Diagnosing Network-Wide Traffic Anomalies," in *Proceedings of ACM SIGCOMM*, Portland, OR, August 2004.
- A. Lakhina, M. Crovella, and C. Diot, "Exploring the Subspace Method for Network-Wide Anomaly Diagnosis," (Poster) in *Network Troubleshooting Workshop: Research, Theory and Operations*, Portland, OR, August 2004.
- A. Lakhina, K. Papagiannaki, M. Crovella, C. Diot, E. D. Kolaczyk, and N. Taft, "Structural Analysis of Network Traffic Flows," in *Proceedings of ACM SIGMETRICS / Performance 2004*, New York, NY, June 2004.
- L. Tang and M. E. Crovella, "A Geometric Exploration of the Landmark Selection Problem," in *Lecture Notes in Computer Science 3015, Proceedings of the 2004 Passive and Active Measurement Workshop (PAM2004)*, pp. 63–72, Juan-les-Pins, France, April 2004.
- L. Tang and M. E. Crovella, "Virtual Landmarks for the Internet," in *Proceedings of the 2003 Internet Measurement Conference*, Miami, Florida, October 2003.
- R. Fonseca, V. Almeida, M. E. Crovella, and B. Abrahão, "On the Intrinsic Locality Properties of Web Reference Streams," in *Proceedings of IEEE Infocom 2003*, San Francisco, California, April 2003.
- A. Lakhina, J. Byers, M. E. Crovella, and P. Xie, "Sampling Biases in IP Topology Measurements," in *Proceedings of IEEE Infocom 2003*, San Francisco, California, April 2003.
- M. E. Crovella and E. Kolaczyk, "Graph Wavelets for Spatial Traffic Analysis," in *Proceedings of IEEE Infocom 2003*, San Francisco, California, April 2003.

- J. Liu, I. Matta, and M. E. Crovella, "End-to-End Inference of Loss Nature in a Hybrid Wired/Wireless Environment," in *Proceedings of Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt'03)*, Sophia-Antipolis, France, March 2003.
- A. Lakhina, J. W. Byers, M. E. Crovella, and Ibrahim Matta, "On the Geographic Location of Internet Resources," (Extended Abstract) in *Proceedings of the ACM SIGCOMM Internet Measurement Workshop 2002*, Marseilles, France, November 2002.
- J. Liu and M. Crovella, "Using Loss Pairs to Discover Network Properties," in *Proceedings of the ACM SIGCOMM Internet Measurement Workshop 2001*, pp. 127–138, San Francisco, CA, November 2001.
- P. Barford, A. Bestavros, J. Byers, and M. Crovella, "On the Marginal Utility of Network Topology Measurements," in *Proceedings of the ACM SIGCOMM Internet Measurement Workshop 2001*, pp. 5–17, San Francisco, CA, November 2001.
- L. Guo, I. Matta, and M. E. Crovella, "How Does TCP Generate Pseudo-Self-Similarity?," in *Proceedings of MAS-COTS 2001*, Cincinnati, Ohio, August 2001.
- P. Barford and M. E. Crovella, "Critical Path Analysis of TCP Transactions," in *Proceedings of the 2000 ACM SIGCOMM Conference*, Stockholm, Sweden, pp. 127–138, September 2000. Also appeared by invitation as "Análisis de secuencia crítica de transacciones TCP" in *Proceedings of SIGCOMM Latin America*, San Antonio de Belén, Costa Rica, April 2001.
- M. E. Crovella, R. Frangioso, and M. Harchol-Balter "Connection Scheduling in Web Servers," in *1999 USENIX Symposium on Internet Technologies and Systems (USITS '99)*.
- P. Barford and M. E. Crovella, "A Performance Evaluation of Hyper Text Transfer Protocols," in *Proceedings of ACM SIGMETRICS '99*, pp. 188–197, Atlanta, Georgia, May 1999.
- A. Bestavros, M. E. Crovella, J. Liu, and D. Martin, "Distributed Packet Rewriting and its Application to Scalable Server Architectures," in *Proceedings of the Sixth International Conference on Network Protocols (ICNP '98)*, pp. 290–297, October, 1998.
- M. Harchol-Balter, M. E. Crovella, and C. D. Murta, "On Choosing a Task Assignment Policy for a Distributed Server System," in *Proceedings of Performance Tools '98, Lecture Notes in Computer Science*, No. 1469, pp. 231–242, September 1998.
- P. Barford and M. E. Crovella, "Generating Representative Web Workloads for Network and Server Performance Evaluation," in *Proceedings of ACM SIGMETRICS/Performance '98*, pp. 151–160, Madison, Wisconsin, July 1998.
- M. E. Crovella, M. Harchol-Balter, and C. D. Murta, "Task Assignment in a Distributed System: Improving Performance by Unbalancing Load," in *Proceedings of SIGMETRICS '98 (Poster paper)*, Madison, Wisconsin, July 1998.
- M. E. Crovella and P. Barford, "The Network Effects of Prefetching," in *Proceedings of IEEE Infocom '98, the Seventeenth Annual Joint Conference of the IEEE Computer and Communication Societies*, San Francisco, California, pp. 1232–1240, April 1998.
- M. E. Crovella and L. Lipsky, "Long-Lasting Transient Conditions in Simulations with Heavy-Tailed Workloads," in *Proceedings of the 1997 Winter Simulation Conference*, December 1997.
- K. Park, G. Kim, and M. E. Crovella, "On the Effect of Traffic Self-Similarity on Network Performance," in *Proceedings of the SPIE International Conference on Performance and Control of Network Systems*, November, 1997.
- P. Fiorini, L. Lipsky, and M. E. Crovella, "Consequences of Ignoring Self-Similar Data Traffic In Communications Modeling," In *Tenth International Conference on Parallel and Distributed Computing Systems (PDCS-97)*, New Orleans, LA, pp. 322–327, October 1997.

- R. L. Carter and M. E. Crovella, "Server Selection using Dynamic Path Characterization in Wide-Area Networks," in *Proceedings of IEEE Infocom '97, the Sixteenth Annual Joint Conference of the IEEE Computer and Communication Societies*, Kobe, Japan, April 1997.
- V. Almeida, A. Bestavros, M. Crovella, and A. de Oliveira, "Characterizing Reference Locality in the WWW," in *Proceedings of the Fourth International Conference on Parallel and Distributed Information Systems (PDIS '96)*, December 1996.
- K. Park, G. Kim, and M. E. Crovella, "On the Relationship Between File Sizes, Transport Protocols, and Self-Similar Network Traffic," in *Proceedings of the Fourth International Conference on Network Protocols (ICNP '96)*, pp. 171–180, October, 1996.
- R. L. Carter and M. E. Crovella, "Measuring Bottleneck Link Speed in Packet-Switched Networks," in *PERFORMANCE '96, the International Conference on Performance Theory, Measurement and Evaluation of Computer and Communication Systems*, October 1996.
- M. E. Crovella and A. Bestavros, "Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes," in *Proceedings of the 1996 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, May 1996. **Winner of a 2010 ACM SIGMETRICS Test of Time Award.**
- M. E. Crovella and R. L. Carter, "Dynamic Server Selection in the Internet," In *Proceedings of the Third IEEE Workshop on the Architecture and Implementation of High Performance Communication Subsystems (HPCS '95)*, August 1995.
- A. Bestavros, R. L. Carter, M. E. Crovella, C. R. Cunha, A. Heddaya, and S. A. Mirdad, "Application-Level Document Caching in the Internet," In *Proceedings of the Second Workshop on Services in Distributed and Networked Environments (SDNE '95)*, June 1995.
- M. E. Crovella and T. J. LeBlanc, "Parallel Performance Prediction Using Lost Cycles Analysis," In *Proceedings of Supercomputing '94*, November, 1994.
- R. Bianchini, M. E. Crovella, L. Kontothanassis, and T. J. LeBlanc, "Software Interleaving," In *Proceedings of the 6th IEEE Symposium on Parallel and Distributed Processing*, October, 1994.
- M. E. Crovella and T. J. LeBlanc, "Performance Debugging using Parallel Performance Predicates," In *3rd ACM/ONR Workshop on Parallel and Distributed Debugging*, May, 1993, pp. 140-150.
- M. Crovella, R. Bianchini, T. LeBlanc, E. Markatos, and R. Wisniewski, "Using Communication-to-Computation Ratio in Parallel Performance Prediction and Program Design," *Proceedings of the Fourth IEEE Symposium on Parallel and Distributed Processing*, December 1992, pp. 238-245.
- M. Crovella, P. Das, C. Dubnicki, T. LeBlanc, and E. Markatos, "Multiprogramming on Multiprocessors," *Proceedings of the Third IEEE Symposium on Parallel and Distributed Processing*, December 1991, pp. 590–597, Also published as TR 385, Department of Computer Science, University of Rochester, February 1991 (revised May 1991).
- E. Markatos, M. Crovella, P. Das, C. Dubnicki, and T. LeBlanc, "The Effects of Multiprogramming on Barrier Synchronization," *Proceedings of the Third IEEE Symposium on Parallel and Distributed Processing*, December 1991, pp. 662-669. Also TR 380, University of Rochester Computer Science Department, May 1991.
- N. Bashir, M. Crovella, G. DeTitta, F. Han, H. Hauptman, J. Horvath, H. King, D. Langs, R. Miller, T. Sabin, P. Thuman and D. Velmurugan, "Parallel Solutions to the Phase Problem in X-Ray Crystallography," *Proceedings of The Fifth Distributed Memory Computing Conference*, IEEE Computer Society Press, 1990, pp. 513–521.
- M. Crovella, M. S. Summers, E. M. Camardo, and S. S. Weissman, "Exploiting a Restricted Application Domain in SDE Design," *Advance Working Papers of the Second International Workshop on Computer Aided Software Engineering*, IEEE Computer Society Press, July 1988, pp. 2-3–2-4.
- J. G. Neal and M. E. Crovella, "An Expert System for Tactical Support Jammer Configuration," *Proceedings of the 1987 IEEE National Aerospace and Electronics Conference (NAECON '87)*, May 1987, pp. 1308–1314.

J. G. Neal, M. E. Crovella, and C. Y. Thielman, "A Knowledge-Based Approach to an EW Mission Planning Task," *Proceedings of the Knowledge Based Information Systems Symposium (KBIS-87)*, SHAPE Technical Centre, The Hague, Netherlands, 1987.

J. G. Neal, D. F. Funke, and M. E. Crovella, "JAMEX: An Expert Advisor on Tactical Support Jammer Configuration," *Proceedings of the Second Annual Conference on Expert Systems in Government*, IEEE Computer Society Press, November 1986, pp. 383–394.

REFEREED PRESENTATIONS

I. Fried, A. Cannistra, C. Casey, A. Piel, T. Lim, M. Crovella, B. Hescott, "CANDL: Coarsely Aligning Networks with Diffusion and Landmarks," Theme Presentation, ISMB/ECCB, Dublin, July 2015.

UNREFEREED PUBLICATIONS

R. Canetti, A. Cohen, C. Conley, M. Crovella, S. L. Dogan, M. Gaboardi, W. Hartzog, R. Van Loo, C. T. Robertson, K. Baird Silbaugh, "National Telecommunications and Information Administration: Comments from Researchers at Boston University and the University of Chicago," Boston Univ. School of Law Research Paper No. 23-24, June 2023.

M. Crovella, "The Skillful Interrogation of the Internet," in *Computer Communication Review*, vol. 49, no. 3, October 2019.

kc Claffy, M. Crovella, T. Friedman, C. Shannon and N. Spring, "Community-oriented network measurement infrastructure (CONMI) workshop report," in *Computer Communication Review*, vol. 36, no. 2, pp. 41–48, April, 2006.

M. E. Crovella, "How Big is the Average Web Page?" in *Inclusions, the Arts of Mathematics*, Journal of the Mathematical Association of America Student Chapter at Boston University, May 1999.

T. J. LeBlanc, M. L. Scott, B. Marsh, E. Markatos, C. Dubnicki, M. Crovella, and T. Becker, "The Psyche Parallel Operating System," *IEEE TC on Operating Systems Newsletter*, vol. 6, no. 1, 1992, pp. 11-13.

TECHNICAL REPORTS

G. Grassi, C. Barakat, M. Crovella, and R. Teixeira, "Leveraging Website Popularity Differences to Identify Performance Anomalies," (Extended Version), Available at <https://hal.inria.fr/hal-03109717>, January 2021.

J. N. Law, N. Tasnina, M. Kshirsagar, J. Klein-Seetharaman, M. Crovella, P. Rajagopalan, S. Kasif, and T. M. Murali, "Identifying Human Interactors of SARS-CoV-2 Proteins and Drug Targets for COVID-19 using Network-Based Label Propagation," arXiv:2006.01968, June 2, 2020.

B. Rastegarpanah, M. Crovella, and K. P. Gummadi, "Fair Inputs and Fair Outputs: the Incompatibility of Fairness in Privacy and Accuracy," arXiv:2005.09209, May 19, 2020.

M.D.M. Leiserson, J. Fan, A. Cannistra, I. Fried, T. Lim, T. Schaffner, M. Crovella, and B. Hescott, "A Multi-Species Functional Embedding Integrating Sequence and Network Structure," bioRxiv, December 4, 2017.

J.-M. Agosta, J. Chandrasekar, M. Crovella, N. Taft, and D. Ting, "Mixture Models of Endhost Network Traffic," arxiv:1212.2744, December 13, 2012.

M. Hacker, M. Crovella, and L. Reyzin, "Secure Pairing of Mobile Devices," BUCS-TR-2012-011, May 16, 2012.

A. Cvetkovski and M. Crovella, "Multidimensional Scaling in the Poincare Disk," arxiv:1105.5332, May 26, 2011.

A. Lakhina, M. Crovella, and C. Diot, "Diagnosing Network-Wide Traffic Anomalies," BUCS-TR-2004-008, February 24, 2004. Also published as a Sprint Research Report.

- A. Lakhina, K. Papagiannaki, M. Crovella, C. Diot, E. D. Kolaczyk, and N. Taft, "Structural Analysis of Network Traffic Flows," BUCS-TR-2003-021, November 10, 2003. Also published as Sprint Research Report RR03-ATL-110708.
- M. Fayed, P. Krapivsky, J. Byers, M. Crovella, and D. Finkel, "On the Size Distribution of Autonomous Systems," BUCS-TR-2003-001, January 17, 2003.
- R. Fonseca, V. Almeida, M. E. Crovella, and B. Abrahão, "On the Intrinsic Locality Properties of Web Reference Streams," BUCS-TR-2002-022, July 15, 2002.
- A. Lakhina, J. Byers, M. E. Crovella, and P. Xie, "Sampling Biases in IP Topology Measurements," BUCS-TR-2002-021, July 15, 2002.
- M. E. Crovella and E. Kolaczyk, "Graph Wavelets for Spatial Traffic Analysis," BUCS-TR-2002-020, July 15, 2002.
- A. Lakhina, J. W. Byers, M. E. Crovella, and Ibrahim Matta, "On the Geographic Location of Internet Resources," BUCS-TR-2002-015, 2002.
- J. Liu, I. Matta, and M. E. Crovella, "End-to-End Inference of Loss Nature in a Hybrid Wired/Wireless Environment," BUCS-TR-2002-008, March 14, 2002.
- M. E. Crovella, R. Frangioso, and M. Harchol-Balter "Connection Scheduling in Web Servers," BUCS-TR-1999-003, March 31, 1999.
- M. Harchol-Balter, M. E. Crovella, and C. D. Murta, "On Choosing a Task Assignment Policy for a Distributed Server System," MIT Laboratory for Computer Science, MIT-LCS-TR-757, 1998.
- P. Barford and M. E. Crovella, "Generating Representative Web Workloads for Network and Server Performance Evaluation," BUCS-TR-1997-006, November 4, 1997.
- M. E. Crovella, M. Harchol-Balter, and C. D. Murta, "Task Assignment in a Distributed System: Improving Performance by Unbalancing Load," BUCS-TR-1997-018, October 31, 1997.
- M. Harchol-Balter, M. E. Crovella, and C. D. Murta, "To Queue or Not to Queue: When Queueing is Better Than Timesharing in a Distributed System," BUCS-TR-1997-017, October 31, 1997.
- K. Park, G. Kim, and M. E. Crovella, "On the Effect of Traffic Self-Similarity on Network Performance," CSD-TR 97-024, Dept. of Computer Sciences, Purdue University, July 1997.
- M. Crovella and P. Barford, "The Network Effects of Prefetching," BUCS-TR-97-002, February 7, 1997.
- K. Park, G. Kim, and M. E. Crovella, "On the Relationship Between File Sizes, Transport Protocols, and Self-Similar Network Traffic," BUCS-TR-96-016, August 7, 1996.
- V. Almeida, A. Bestavros, M. Crovella, and A. deOliveira, "Characterizing Reference Locality in the WWW," BUCS-TR-96-011, June 21, 1996.
- A. Bestavros, M. Chen, M. Crovella, A. Heddaya, S. Sclaroff, and J. Cowie, "Responsive Web Computing: Resource Management, Protocol Techniques, and Applications (A research statement)," BUCS-TR-96-008, March 21, 1996.
- R. L. Carter and M. E. Crovella, "Dynamic Server Selection Using Bandwidth Probing in Wide-Area Networks," BUCS-TR-96-007, Boston University Computer Science Department, March 18, 1996.
- R. L. Carter and M. E. Crovella, "Measuring Bottleneck Link Speed in Packet-Switched Networks," Technical Report BUCS-TR-96-006, Boston University Computer Science Department, March 15, 1996.
- M. E. Crovella and A. Bestavros, "Explaining World Wide Web Traffic Self-Similarity," Technical Report BUCS-TR-95-015 (Revised), Boston University Department of Computer Science, October 1995.
- M. E. Crovella, T. J. LeBlanc, and W. Meira Jr., "Performance Measurement and Modeling with the Lost Cycles Toolkit," Technical Report 580, University of Rochester Computer Science Department, June 1995.

- C. Cunha, A. Bestavros, and M. Crovella, "Characteristics of WWW Client-based Traces," Technical Report BUCS-TR-95-010, Boston University Department of Computer Science, April 1995.
- A. Bestavros, R. L. Carter, M. E. Crovella, C. R. Cunha, A. Heddaya, and S. A. Mirdad, "Application-Level Document Caching in the Internet," Technical Report BUCS-TR-95-002, Boston University Department of Computer Science, February 1995.
- M. E. Crovella, "Performance Prediction and Tuning of Parallel Programs," Technical Report 573 and Ph.D. Thesis, Computer Science Dept., U. Rochester, February 1995.
- M. E. Crovella and T. J. LeBlanc, "The Search for Lost Cycles: a New Approach to Parallel Program Performance Evaluation," Technical Report 479, University of Rochester Computer Science Department, December 1993.
- L. A. Crawl, M. E. Crovella, T. J. LeBlanc, and M. L. Scott, "Beyond Data Parallelism: The Advantages of Multiple Parallelizations in Combinatorial Search," Technical Report 92-80-06, December 1992, Department of Computer Science, Oregon State University. Revised version appears as Technical Report 451, U. Rochester Department of Computer Science, April 1993.
- R. Bianchini, M. E. Crovella, L. Kontothanassis, and T. J. LeBlanc, "Alleviating Memory Contention in Matrix Computations on Large-Scale Shared-Memory Multiprocessors," Technical Report 449, U. Rochester Computer Science Department, April 1993.
- R. Bianchini, M. E. Crovella, L. Kontothanassis, and T. J. LeBlanc, "Memory Contention in Scalable Cache-Coherent Multiprocessors," Technical Report 448, U. Rochester Computer Science Department, April 1993.
- D. Bergmark and M. Crovella, "Case Study in KSR Programming: Finding Outliers by the Minimum Volume Ellipsoid Method," Technical Report CTC92TR107, October 1992, Center for Theory and Simulation in Science and Engineering, Cornell University, Ithaca, NY 14853.

RECENT TALKS

"Beyond accuracy: understanding the performance of LLMs on exams designed for humans,"

- UFMG, Department of Computer Science, March 8, 2024.

"Estimating Traffic Flows that Can't Be Measured"

- Google Telemetry Seminar Series, May 17, 2021.

"Assessing Candidate Preference Through Web Browsing History"

- Sorbonne Université (UPMC) LIP6, April 15, 2019.
- INRIA Sophia-Antipolis, March 20, 2019.
- INRIA Saclay, December 13, 2018.
- INRIA Paris, November 8, 2018.
- LINC Laboratory, Paris, October 10, 2018.

"Human Behavior is Low Dimensional"

- TMA Experts Summit 2019, Paris, June 18, 2019.
- Chinese Academy of Sciences, Institute of Computing Technology, Beijing, May 20, 2019.
- Huawei Network AI Workshop, Paris, April 9, 2019.
- LINC Laboratory, Paris, February 13, 2019.
- Wesleyan University Department of Mathematics and Computer Science, April 13, 2017.
- IEEE Workshop on Data Science for Networking (DS4N), December 14, 2017.

"The Statistics of Human Activity and Implications for Security"

- BU Statistics Club, Professor Speaker Series, November 10, 2016.
- “Analyzing the Evolution of Internet Routing through Path Analysis”
- Cisco FAST Seminar Series, July 17, 2015.
 - Raytheon BBN Technologies, SDP Seminar, September 28, 2015.
- “Reversing the Curse of Dimensionality for Security Analysis”
- NYU Poly Department of Computer Science, December 5, 2014.
- “Anomaly Detection and Dimensionality Reduction”
- Workshop on Mathematics of Data Analysis in Cybersecurity, ICERM, Brown University, October 24, 2014.
- “Targeted Matrix Completion”
- 51st Annual Allerton Conference on Communication, Control, and Computing, University of Illinois at Urbana-Champaign, October 3, 2013.
- “Better Routing with Hyperbolic Geometry”
- Laboratory of Information, Networking and Communication Sciences (LINCS) Paris, December 19, 2012.
- “A Fine Grained Distance Metric for Small Worlds”
- Université de Liège, Liège Belgium, January 11, 2013.
 - Laboratory of Information, Networking and Communication Sciences (LINCS) Paris, Mar 20, 2012.
- “Inferring Invisible Traffic”
- Williams College, Dept. of Computer Science, Williamstown, MA, November 22, 2013.
 - Tufts University, Dept. of Computer Science, Medford, MA, November 7, 2013.
 - University of Miami, Dept. of Computer Science, Miami, FL, March 2, 2011.
 - International Computer Science Institute (ICSI), Berkeley, CA, July 20, 2010.
 - Statistics of Networks Workshop, Isaac Newton Institute for Mathematical Sciences, Cambridge, UK, June 24, 2010.
- “Network Monitoring: Challenges and Opportunities”
- Indian Institute of Technology (IIT), Delhi, India, June 9, 2008.
- “Exploiting Feature Distributions in Anomaly Diagnosis”
- Dartmouth, November 28, 2005.
 - MIT CSAIL-LIDS Networking and Systems Seminar, Cambridge MA, April 11, 2005.
- “Network Anomaly Diagnosis”
- Boston Area Networking and Networkers Annual Summit (BANANAS-2005), BBN Technologies, Cambridge MA, January 20, 2005.
- “Applying the Subspace Method to Network Traffic Analysis”
- Statistical Research Center for Complex Systems (SRCCS), Workshop on “Internet Measurement, Modeling, and Analysis,” Seoul National University, Seoul, South Korea, January 11, 2005.
 - Electronics and Telecommunications Research Institute (ETRI), Daejeon, South Korea, January 13, 2005.
- “Diagnosing Network-Wide Traffic Anomalies”

- Columbia University, Nov. 17, 2004.
- Workshop on Internet Signal Processing (WISP), CAIDA, Nov. 11, 2004.
- Ecole Normale Supérieure, Paris, Apr. 13, 2004.
- France Telecom R&D, Paris, Mar. 18, 2004.

“Structural Analysis of Network Traffic Flows”

- Project Metropolis Meeting, École Nationale Supérieure des Télécommunications de Bretagne (ENST), Rennes, Feb. 10, 2004.
- Institute for Mathematics and its Applications, Workshop on “Measurement, Modeling and Analysis of the Internet,” Minneapolis, MN, January 12, 2004.

“Virtual Landmarks for the Internet”

- Intel Research, Cambridge UK, October 1, 2003.
- Ecole Polytechnique Fédérale de Lausanne (EPFL), November 10, 2003.
- Laboratoire d’Informatique de Paris VI (LIP6), November 14, 2003.

“Sampling Biases in IP Topology Measurements”

- Séminaire Graphes, Réseaux et Modélisation, Laboratoire de Recherche en Informatique, Université d’Orsay, January 8, 2004.
- IBM T.J. Watson Labs, April 11, 2003.
- BU Statistics Colloquium, April 10, 2003.
- National Institute of Statistical Sciences, March 28, 2003.

Appendix 2: Materials Considered in the Preparation of this Declaration

Exhibit No.	Description
1001	U.S. Patent No. 8,995,693 to Boldyrev et al. (“the ’693 patent”)
1002	Prosecution History for the ’693 patent
1004	Foster et al., <i>IBM InfoSphere Streams: Assembling Continuous Insight in the Information Revolution</i> (2011) (“Foster”)
1005	Williams et al., <i>Implementing CICS Web Services</i> (2007) (“Williams”)
1006	Declaration of N. E. Frank-White re Foster (Ex-1004)
1007	Declaration of N. E. Frank-White re Williams (Ex-1005)
1008	Declaration of M. Spicer re Foster (Ex-1004)
1009	Declaration of M. Ebbers re Williams (Ex-1005)
1010	U.S. Patent App. Pub. No. 2008/0127064A1 to Orofino et al. (“Orofino”)
1011	Kumar et al., <i>DEDUCE: At the Intersection of MapReduce and Stream Processing</i> (2010) (“Kumar”)
1012	Verma et al., <i>ARIA: Automatic Resource Inference and Allocation for MapReduce Environments</i> (2011) (“Verma”)
1013	7 Blackwell Publ’g, <i>The Blackwell Encyclopedia of Management, Management Information Systems</i> (Davis ed., 2d ed. 2005) (excerpts)
1014	U.S. Patent. App. Pub. No. 2012/0083715A1 (“Yuen”)
1015	O’Reilly & Associates, <i>JAVA Enterprise in a Nutshell</i> (2001) (“O’Reilly”)
1016	U.S. Patent. App. Pub. No. 2004/0031037A1 to Ikoma et al. (“Ikoma”)
1017	Crockford, <i>Request for Comments 4627</i> (2006) (“RFC4627”)

Appendix 3: Challenged Claims

1[pre]. A method comprising:

[1a] determining via a processor at least one processing element of at least one dynamic processing mechanism;

[1b] causing, at least in part, a marshalling via the processor of the at least one processing element as at least one data object;

[1c] transferring of the at least one data object to at least one static processing mechanism;

[1d] processing of the at least one data object by the at least one static processing mechanism; and

[1e] after processing, transferring the at least one data object back to the at least one dynamic processing mechanism.

2[a]. The method of claim 1, further comprising:

a processing of at least substantially real-time data via the at least one processing element to determine one or more triggering events,

[2b] wherein, the transferring of the at least one data object to the at least one static processing mechanism is based, at least in part, on the one or more triggering events, and

[2c] wherein the one or more triggering events include, at least in part, one or more contextual trends occurring above at least one threshold level.

3. The method of claim 2, further comprising: unmarshalling of the at least one processing element as the at least one data object.

4. The method of claim 2, wherein the processing of the at least one data object via the at least one static processing mechanism causes, at least in part, an updating of the at least one data object based, at least in part, on the one or more triggering events, the one or more contextual trends, or a combination thereof.

5[a]. The method of claim 4, further comprising: at least one determining of one or more computational chains, one or more parameters of the one or more computational chains, one or more data elements of the one or more computational chains, or a combination thereof associated with the at least one processing element,

[5b] wherein at least one data object includes, at least in part, the one or more computational chains, the one or more parameters, the one or more data elements, or a combination thereof.

7. The method of claim 1, wherein the processing the processing of the at least one data object causes, at least in part, updating of one or more components of the at least one static processing mechanism.

8[a]. The method of claim 1, aggregating of one or more outputs of the at least one dynamic processing mechanism and the one or more other outputs of the at least one static processing mechanism; and

[8b] at least one determination of one or more outputs of at least one processing job based, at least in part, on the aggregating.

9. The method of claim 1, wherein the at least one static processing mechanism operates, at least in part, in a batch mode with a predetermined batch frequency; and wherein the at least one dynamic processing mechanism, the at least one processing element, or a combination thereof operate, at least in part, in substantially real-time or at a frequency greater than the predetermined batch frequency.

10. The method of claim 1, wherein the at least one static processing mechanism is associated with performing, at least in part, slow moving data analytics; and wherein the at least one dynamic processing mechanism, the at least one processing element, or a combination thereof are associated with performing, at least in part, fast moving data analytics.

11[pre]. An apparatus comprising:

[11a] at least one processor; and

at least one memory including computer program code for one or more programs,

the at least one memory and the computer program code configured to, with the at least one processor, cause the apparatus to perform at least the following,

[11b] determine at least one processing element of at least one dynamic processing mechanism;

[11c] cause a marshalling via the processor of the at least one processing element as at least one data object;

[11d] transfer of the at least one data object to at least one static processing mechanism;

[11e] process the at least one data object by the at least one static processing mechanism; and

[11f] after the process, transfer the at least one data object back to the at least one dynamic processing mechanism.

12[a]. The apparatus of claim 11, wherein the apparatus is further caused to: process and/or facilitate a processing of at least substantially real-time data via the at least one processing element to determine one or more triggering events,

[12b] wherein the transfer of the at least one data object to the at least one static processing mechanism based, at least in part, on the one or more triggering events, and

[12c] wherein the one or more triggering events include, at least in part, one or more contextual trends occurring above at least one threshold level.

13. The apparatus of claim 12, wherein the apparatus is further caused to: cause, at least in part, an unmarshalling of the at least one processing element as at least one data object.

14. The apparatus of claim 12, wherein the process of the at least one data object via the at least one static processing mechanism causes, at least in part, an updating of the at least one data object based, at least in part, on the one or more triggering events, the one or more contextual trends, or a combination thereof.

15[a]. The apparatus of claim 14, wherein the apparatus is further caused to: determine one or more computational chains, one or more parameters of the one or more computational chains, one or more data elements of the one or more computational chains, or a combination thereof associated with the at least one processing element,

[15b] wherein at least one data object includes, at least in part, the one or more computational chains, the one or more parameters, the one or more data elements, or a combination thereof.

17. The apparatus of claim 11, wherein process of the at least one data object causes, at least in part, an updating of one or more components of the at least one static processing mechanism.

18[a]. The apparatus of claim 11, wherein the apparatus is further caused to:

cause, at least in part, an aggregation of one or more outputs of the at least one dynamic processing mechanism and the one or more other outputs of the at least one static processing mechanism; and

[18b] determining one or more outputs of at least one processing job based, at least in part, on the aggregation.

19. The apparatus of claim 11, wherein the at least one static processing mechanism operates, at least in part, in a batch mode with a predetermined batch frequency; and wherein the at least one dynamic processing mechanism, the at least one processing element, or a combination thereof operate, at least in part, in substantially real-time or at a frequency greater than the predetermined batch frequency.

20. The apparatus of claim 11, wherein the at least one static processing mechanism is associated with performing, at least in part, slow moving data analytics; and wherein the at least one dynamic processing mechanism, the at least one processing element, or a combination thereof are associated with performing, at least in part, fast moving data analytics.

Appendix 4: Understanding of the Law

I have applied the following legal principles provided to me by counsel in arriving at the opinions set forth in this report.

Legal Standard for Prior Art

I am not an attorney. I have been informed by attorneys of the relevant legal principles and have applied them to arrive at the opinions set forth in this declaration.

I understand that the petitioner for inter partes review may request the cancelation of one or more claims of a patent based on grounds available under 35 U.S.C. § 102 and 35 U.S.C. § 103 using prior art that consists of patents and printed publications.

Anticipation and Prior Art

I understand that § 102 specifies when a challenged claim is invalid for lacking novelty over the prior art, and that this concept is also known as “anticipation.” I understand that a prior art reference anticipates a challenged claim, and thus renders it invalid by anticipation, if all elements of the challenged claim are disclosed in the prior art reference. I understand the disclosure in the prior art reference can be either explicit or inherent, meaning it is necessarily present or implied. I understand that the prior art reference does not have to use the same words as the challenged claim, but all of the requirements of the claim

must be disclosed so that a person of ordinary skill in the art could make and use the claimed subject-matter.

In addition, I understand that § 102 also defines what is available for use as a prior art reference to a challenged claim.

Under § 102(a), a challenged claim is anticipated if it was patented or described in a printed publication in the United States or a foreign country before the challenged claim's date of invention.

Under § 102(b), a challenged claim is anticipated if it was patented or described in a printed publication in the United States or a foreign country more than one year prior to the challenged patent's filing date.

Under § 102(e), a challenged claim is anticipated if it was described in a published patent application that was filed by another in the United States before the challenged claim's date of invention, or was described in a patent granted to another that was filed in the United States before the challenged claim's date of invention.

I understand that a challenged claim's date of invention is presumed to be the challenged patent's filing date. I also understand that the patent owner may establish an earlier invention date and "swear behind" prior art defined by § 102(a) or § 102(e) by proving (with corroborated evidence) the actual date on which the

named inventors conceived of the subject matter of the challenged claim and proving that the inventors were diligent in reducing the subject matter to practice.

I understand that the filing date of a patent is generally the filing date of the application filed in the United States that issued as the patent. However, I understand that a patent may be granted an earlier effective filing date if the patent owner properly claimed priority to an earlier patent application.

I understand that when a challenged claim covers several structures, either generically or as alternatives, the claim is deemed anticipated if any of the structures within the scope of the claim is found in the prior art reference.

I understand that when a challenged claim requires selection of an element from a list of alternatives, the prior art teaches the element if one of the alternatives is taught by the prior art.

Legal Standard for Obviousness

I understand that even if a challenged claim is not anticipated, it is still invalid if the differences between the claimed subject matter and the prior art are such that the claimed subject matter would have been obvious to a person of ordinary skill in the pertinent art at the time the alleged invention.

I understand that obviousness must be determined with respect to the challenged claim as a whole.

I understand that one cannot rely on hindsight in deciding whether a claim is obvious.

I also understand that an obviousness analysis includes the consideration of factors such as (1) the scope and content of the prior art, (2) the differences between the prior art and the challenged claim, (3) the level of ordinary skill in the pertinent art, and (4) “secondary” or “objective” evidence of non-obviousness.

Secondary or objective evidence of non-obviousness includes evidence of: (1) a long felt but unmet need in the prior art that was satisfied by the claimed invention; (2) commercial success or the lack of commercial success of the claimed invention; (3) unexpected results achieved by the claimed invention; (4) praise of the claimed invention by others skilled in the art; (5) taking of licenses under the patent by others; (6) deliberate copying of the claimed invention; and (7) contemporaneous and independent invention by others. However, I understand that there must be a relationship between any secondary evidence of non-obviousness and the claimed invention.

I understand that a challenged claim can be invalid for obviousness over a combination of prior art references if a reason existed (at the time of the alleged invention) that would have prompted a person of ordinary skill in the art to combine elements of the prior art in the manner required by the challenged claim. I understand that this requirement is also referred to as a “motivation to combine,”

“suggestion to combine,” or “reason to combine,” and that there are several rationales that meet this requirement.

I understand that the prior art references themselves may provide a motivation to combine, but other times simple common sense can link two or more prior art references. I further understand that obviousness analysis recognizes that market demand, rather than scientific literature, often drives innovation, and that a motivation to combine references may come from market forces.

I understand obviousness to include, for instance, scenarios where known techniques are simply applied to other devices, systems, or processes to improve them in an expected or known way. I also understand that practical and common-sense considerations should be applied in a proper obviousness analysis. For instance, familiar items may have obvious uses beyond their primary purposes.

I understand that the combination of familiar elements according to known methods is obvious when it yields predictable results. For instance, obviousness bars patentability of a predictable variation of a technique even if the technique originated in another field of endeavor. This is because design incentives and other market forces can prompt variations of it, and predictable variations are not the product of innovation, but rather ordinary skill and common sense.

I understand that a particular combination may be obvious if it was obvious to try the combination. For example, when there is a design need or market

pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. This would result in something obvious because the result is the product not of innovation but of ordinary skill and common sense. However, I understand that it may not be obvious to try a combination when it involves unpredictable technologies.

It is further my understanding that a proper obviousness analysis focuses on what was known or obvious to a person of ordinary skill in the art, not just the patentee. Accordingly, I understand that any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed.

It is my understanding that the Manual of Patent Examining Procedure §2143 sets forth the following as exemplary rationales that support a conclusion of obviousness:

- Combining prior art elements according to known methods to yield predictable results;
- Simple substitution of one known element for another to obtain predictable results;
- Use of known technique to improve similar devices (methods, or products) in the same way;

- Applying a known technique to a known device (method, or product) ready for improvement to yield predictable results;
- Choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success;
- Known work in one field of endeavor may prompt variations of it for use in either the same field or a different one based on design incentives or other market forces if the variations are predictable to one of ordinary skill in the art;
- Some teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill to modify the prior art reference or to combine prior art reference teachings to arrive at the claimed invention.

A person of ordinary skill in the art looking to overcome a problem will often use the teachings of multiple publications together like pieces of a puzzle, even though the prior art does not necessarily fit perfectly together. Therefore, I understand that references for obviousness need not fit perfectly together like puzzle pieces. Instead, I understand that obviousness analysis takes into account inferences, creative steps, common sense, and practical logic and applications that a person of ordinary skill in the art would employ under the circumstances.

I understand that a claim can be obvious in light of a single reference, if the elements of the challenged claim that are not explicitly or inherently disclosed in the reference can be supplied by the common sense of one of skill in the art.

I understand that obviousness also bars the patentability of applying known or obvious design choices to the prior art. One cannot patent merely substituting one prior art element for another if the substitution can be made with predictable results. Likewise, combining prior art techniques that are interoperable with respect to one another is generally obvious and not patentable.

In order for a claim to be found invalid based upon a modification or combination of the prior art, there must be reasonable expectation that a person of ordinary skill would have successfully modified or combined the prior art to arrive at the claimed arrangement. This does not mean that it must be certain that a person of ordinary skill would have been successful – the law only requires that the person of ordinary skill in the art would have perceived a reasonable expectation of success in modifying or combining the prior art to arrive at the claimed invention.

In sum, my understanding is that obviousness invalidates claims that merely recite combinations of, or obvious variations of, prior art teachings using understanding and knowledge of one of skill in the art at the time and motivated by the general problem facing the inventor at the time. Under this analysis, the prior art references themselves, or any need or problem known in the field of endeavor

at the time of the invention, can provide a reason for combining the elements of or attempting obvious variations on prior art references in the claimed manner.

Legal Standard for Claim Construction

I understand that before any invalidity analysis can be properly performed, the scope and meaning of the challenged claims must be determined by claim construction.

I understand that a patent may include two types of claims, independent claims and dependent claims. I understand that an independent claim stands alone and includes only the limitations it recites. I understand that a dependent claim depends from an independent claim or another dependent claim. I understand that a dependent claim includes all the limitations that it recites in addition to the limitations recited in the claim (or claims) from which it depends.

In comparing the challenged claims to the prior art, I have carefully considered the patent and its file history in light of the understanding of a person of skill at the time of the alleged invention.

I understand that to determine how a person of ordinary skill would have understood a claim term, one should look to sources available at the time of the alleged invention that show what a person of skill in the art would have understood disputed claim language to mean. It is my understanding that this may include what is called “intrinsic” evidence as well as “extrinsic” evidence.

I understand that, in construing a claim term, one should primarily rely on intrinsic patent evidence, which includes the words of the claims themselves, the remainder of the patent specification, and the prosecution history. I understand that extrinsic evidence, which is evidence external to the patent and the prosecution history, may also be useful in interpreting patent claims when the intrinsic evidence itself is insufficient. I understand that extrinsic evidence may include principles, concepts, terms, and other resources available to those of skill in the art at the time of the invention.

I understand that words or terms should be given their ordinary and accepted meaning unless it appears that the inventors were using them to mean something else or something more specific. I understand that to determine whether a term has special meaning, the claims, the patent specification, and the prosecution history are particularly important, and may show that the inventor gave a term a particular definition or intentionally disclaimed, disavowed, or surrendered claim scope.

I understand that the claims of a patent define the scope of the rights conferred by the patent. I understand that because the claims point out and distinctly claim the subject matter which the inventors regard as their invention, claim construction analysis must begin with and is focused on the claim language itself. I understand that the context of the term within the claim as well as other claims of the patent can inform the meaning of a claim term. For example, because

claim terms are normally used consistently throughout the patent, how a term is used in one claim can often inform the meaning of the same term in other claims. Differences among claims or claim terms can also be a useful guide in understanding the meaning of particular claim terms.

I understand that a claim term should be construed not only in the context of the particular claim in which the disputed term appears, but in the context of the entire patent, including the entire specification. I understand that because the specification is a primary basis for construing the claims, a correct construction must align with the specification.

I understand that the prosecution history of the patent as well as art incorporated by reference or otherwise cited during the prosecution history are also highly relevant in construing claim terms. For instance, art cited by or incorporated by reference may indicate how the inventor and others of skill in the art at the time of the invention understood certain terms and concepts. Additionally, the prosecution history may show that the inventors disclaimed or disavowed claim scope, or further explained the meaning of a claim term.

With regard to extrinsic evidence, I understand that all evidence external to the patent and prosecution history, including expert and inventor testimony, dictionaries, and learned treatises, can also be considered. For example, technical dictionaries may indicate how one of skill in the art used or understood the claim

terms. However, I understand that extrinsic evidence is considered to be less reliable than intrinsic evidence, and for that reason is generally given less weight than intrinsic evidence.

I understand that in general, a term or phrase found in the introductory words or preamble of the claim, should be construed as a limitation if it recites essential structure or steps, or is necessary to give meaning to the claim. For instance, I understand preamble language may limit claim scope: (i) if dependence on a preamble phrase for antecedent basis indicates a reliance on both the preamble and claim body to define the claimed invention; (ii) if reference to the preamble is necessary to understand limitations or terms in the claim body; or (iii) if the preamble recites additional structure or steps that the specification identifies as important.

On the other hand, I understand that a preamble term or phrase is not limiting where a challenged claim defines a structurally complete invention in the claim body and uses the preamble only to state a purpose or intended use for the invention. I understand that to make this determination, one should review the entire patent to gain an understanding of what the inventors claim they invented and intended to encompass in the claims.

I understand that 35 U.S.C. § 112 ¶ 6 created an exception to the general rule of claim construction called a “means plus function” limitation. These types of

terms and limitations should be interpreted to cover only the corresponding structure described in the specification, and equivalents thereof. I also understand that a limitation is presumed to be a means plus function limitation if (a) the claim limitation uses the phrase “means for”; (b) the “means for” is modified by functional language; and (c) the phrase “means for” is not modified by sufficient structure for achieving the specified function.

I understand that a structure is considered structurally equivalent to the corresponding structure identified in the specification only if the differences between them are insubstantial. For instance, if the structure performs the same function in substantially the same way to achieve substantially the same result. I further understand that a structural equivalent must have been available at the time of the issuance of the claim.