Informatics Challenges in Future Internet

Keith G Jeffery
Director International Relations, Science and Technology Facilities Council, UK
President-Elect ERCIM
keith.jeffery@stfc.ac.uk

Introduction

A great advantage of age is the wealth of experience gained over the years. Commonly mature researchers dismiss the latest research as a re-hash of something done before. In fact usually modern research is a re-use or re-interpretation with additional elaboration on consolidated research results obtained earlier. For many the concepts of Future Internet (FI), personal mobile devices, GRIDs, CLOUDs, Web 2.0, social networking or even WWW itself are all seen as ‘hype’ or old concepts or even products wrapped in ‘new clothes’. There is some truth in this, but each of these technological components of FI either integrated previous technologies in a new way, extended pre-existing technologies to make them more usable or allied the technology to a new business model so increasing accessibility / availability.

Unfortunately, below the ‘hype’ level of enthusiastic discussion on FI and its benefits, lie many unsolved technological problems (social, legal and economic problems are not addressed here except occasionally when co-related to a technological challenge). A discussion of those challenges forms the core of this paper.

The paper first discusses and characterises FI, then surveys the current state of ICT and finally deals with the technological challenges.

Future Internet

The term ‘future internet’ (FI) is basically the idea of a society where people, objects, services and information sources are completely connected and available to end-users through any appropriately-configured device. It may be argued there is no FI since this is a continuing evolution of technology and societal demands. Those of us who remember attempts in the 1960s and 1970s to connect computers to each other, connect them to detector devices for data gathering and to connect interactive end-user devices, will note that many of the problems identified then still exist. The difference is the

magnitude with huge numbers of devices or appliances having IP addresses in the FI. However FI is a useful label for the emerging configuration of ICT.
In a European context the EC has initiated a 600m€ Future Internet PPP (Public-Private Partnership) activity with a General Assembly and funded projects; however the activity has been dominated by commercial companies pursuing agendas for their own purposes and in many ways is lacking the innovation required to ensure that the expected societal benefits especially in the quality of life (security, health environment, education, entertainment, transport) are realised alongside the wealth-creation benefits.

In many ways FI is an integration or packaging of the results of earlier research and development.

At the base is networking. Provision of a 100Gb or faster backbone, with widely available fast inexpensive wireless zones, is a prerequisite and the increasing number of devices will require a greater number of IP addresses and better scheduling and utilisation of bandwidth. IPv6 is intended to accomplish this. Connections to the network devices are standardised with appropriate physical and logical connections for devices of different capabilities. However, is should be noted that – over the last 20 years – improvements in performance and in cost-effectiveness of networking have lagged by many orders of magnitude those achieved in data storage and processors. It would appear that the problem is not technology, but the willingness of telecommunications suppliers to make available appropriate bandwidth.

Connected to the network are various devices: for end-users, for detection / data collection, for control, for processing and for storage. There have been great improvements in performance, miniaturisation and packaging, power consumption, flexibility and adaptability and mobility. There has been less progress on heat output and its management including utilisation. There is little standardisation in the devices themselves leading to a rich diversity and choice but – in general – required hardware interfaces are standardised allowing some degree of ‘plug and play’.

The devices depend on operating systems which control the resources available and provide the interconnections. Here there are strong proprietary influences at the user device level (Microsoft, Apple, Symbian, Android etc for mobiles with Windows and various versions of Linux for laptops) necessarily due to the differing characteristics of the hardware being controlled. Interestingly, tablet devices are classified with mobile phones rather than laptops – essentially because they lack backing (disk) storage.

However the operating systems generally adopt communication protocols that are widely used so providing a base-level of interoperation. There is a real challenge in cooperative control of heterogeneous devices in a distributed environment because of the varying characteristics of the operating system – and particularly its local configuration - of each device.

Software standards exist but commonly proprietary features are added by suppliers - for commercial advantage - making interoperation a task of identifying the most feature-rich subset of characteristics available on all platforms.

Data standards are in many ways the key. The formal standardisation of Unicode for character sets was a significant step (although many legacy systems exist which are not Unicode-compatible). Above this there are many standards at both syntactic and semantic levels, usually application-dependent. However some general standards – such as those developed by W3C (World Wide Web Consortium) of which XML is an example – provide a general basis for syntactic interoperability. Semantic interoperability requires facilities to enable understanding – by computer systems – of the meaning of terms. W3C has standardised OWL and SKOS but, in fact, any system capable of storing triples provides the logic basis for semantic interoperability.

In many ways FI is based on the concept of e-Science articulated first in 1999 in UK [Je99] (itself an integration of the results of much earlier thinking on systems and e-infrastructure) and developed much further leading to a summary as 'the fourth paradigm' [HeTaTo2009]. The key principles of (a) virtualisation to the degree where the end-user neither knows nor cares where the data is stored, where the processing is done, how the resources are obtained and (b) how the connection is maintained as long as appropriate service level is achieved were developed within the e-Science concept utilising first GRIDs and later CLOUDs as configurations to provide this virtualisation.

The key technological demands of FI include (a) very high speed access; (b) multiple functional devices for processing, storage, detection, control, user access requirements with standard network interconnects and appropriate performance; (c) interoperation of software; (d) interoperation of data and information; (e) security and privacy appropriate to the user and the application. In this paper we concentrate on key technologies required to realise the FI and particularly (1) metadata; (2) state; (3) data representativity; (4) data quality, veracity and permanence; (5) trust, security and privacy; (6) management of service levels and quality of service; (7) systems design, development, operation and decommissioning. It is no surprise that in examining these technologies we challenge conventional, existing ICT and the underlying computer science.

Existing ICT

Existing ICT depends dominantly on the concepts of Alan Turing (the Turing machine) [Tu36] which implies sequential processing and of John von Neumann (stored programme hardware / operating system architecture) [vonNe45] which optimises for expensive CPU and memory resources. These concepts were developed in an era when interconnection of computing devices was not envisaged.

Much modern existing ICT falls into the following main categories:

1. Large scale transaction processing for line-of-business applications (including production industry) with associated data warehouses for analytics and decision support. This is typical of most industry, commercial and government organisations and major products are available from e.g. ORACLE and SAP;
2. CAD/CAM (Computer-aided design, computer-aided manufacturing) systems for production industry especially including the use of production robots;
3. Control and optimisation systems for production industry (e.g. oil refinery, chemical plant) or for management of air traffic, water supply, electricity grid, transport logistics etc;
4. Scientific / engineering systems involving data collection from observation or experiment, simulation and complex analysis and visualisation. Data collection commonly requires managing streamed data. This is typical of enterprises in science, engineering and research;
5. Systems to access, update, retrieve from multiple multimedia information sources (sometimes resolving linguistic, syntactic and semantic heterogeneity) with general or subject domain scale often interlinked with the above;
6. Systems for multimodal human intercommunication with voice, e-mail, social networking, games and education, documents (creation, editing, sharing and retrieval), audio, video, task lists, calendaring and associated push technologies usually connected with the above but sometimes used independently;

The majority of these kinds of systems utilise WWW at the user interface and structured relational databases ‘behind the scenes’ commonly in B2C

(Business to Customer) applications whether within a company or public-facing (such online purchasing or online banking). Additionally B2B (Business to Business) utilising relational databases and web services is used commonly for many applications especially supply chain environments. Many examples of such systems are meeting problems in performance, reliability / business continuity, security, privacy and – where choice is involved – take-up by end-users.

Examples of systems envisaged in the FI – and differentiated from existing ICT systems which will continue and evolve - include:

a) Systems for healthcare at home; monitoring, measuring, advising and administering appropriate drugs;
b) Systems integrating home refrigerator stock control with online delivery of foodstuffs;
c) Systems assisting a traveller in completing their journey whether by public or private car, train, boat, plane with appropriate factors such as time, cost, safety, convenience, environmental cost;
d) Systems for self-paced life-long education and training (including interactions);
e) Systems for managing energy use in homes and industrial / commercial buildings with appropriate factors such as cost, safety, legislation protecting rights of humans, health advice;
f) Systems for politically democratic access to information, debate, discussion and voting;
g) Systems for advanced distributed multiplayer games;
h) Systems for barter and exchange of goods and services;
i) Systems for integrated urban planning including services as well as buildings and physical infrastructure;

These examples require high performance, adaptable, user-friendly systems integrating communications, detectors, processing and data storage within an appropriate framework of service levels covering performance, reliability, trust, security, privacy and legislation.

Put simply the requirement is - instead of imposing ERP (Enterprise Resource Planning)-type systems on end-users (whether within an organisation or as customers or suppliers outside it) - constructing user-centric, flexible systems dynamically to meet the current (and about to change) end-user needs.

The Challenges

To move with success from the existing ICT environments to the FI requires some key developments, most of which do not yet have solutions (or adequate solutions) and most of which require extensive research to reach a suitable state of development.

Metadata

Metadata is data about data. What is metadata to one application is data to another. Thus metadata provides a logical view for utilisation of some data. Since metadata is critically important for interoperation and semantic understanding, there is a requirement for precise and formal representation of metadata to allow automated processing. A basic classification was proposed [Je00] distinguishing schema, navigational and associative metadata, the latter partitioned into descriptive (like a library catalogue card), restrictive (covering e.g. copyright or charges for use) and supportive (relating to the whole domain and providing thesauri, domain ontologies). Research is required into the metadata representation language expressivity in order to represent the real world accurately. For example, the existing Dublin Core Metadata standard [DC] is machine-readable but not machine-understandable, and furthermore mixes navigational, associative descriptive and associative restrictive metadata. A formal version has been proposed [Je99a]. However, metadata is now a 'hot' topic and there is a plethora of 'standards' in the area with consequent problems for interoperation and utilisation of already-collected datasets.

State

Computer systems are expected to portray accurately the state of the real world of interest. Air traffic control systems are expected to provide – on a second-by-second basis - an accurate picture of the location, height, speed and type of planes. Nuclear reactor control systems similarly should portray the state of the reactor including all pertinent variables. In a conventional database management system state is maintained in near-real time by ACID (Atomicity, Consistency, Isolation, Durability) transactions which lock areas of the database to prevent update (or repeatable read) conflicts. As transactions become more complex in such an environment (more instructions related to the update, more database tables affected) the duration of locking – with subsequent performance degradation - becomes unacceptable and so other techniques are used. Roll-back assumes that conflicts occur rarely and so to improve performance allows transactions to proceed without locking with the possibility that a conflict occurs leaving the database in an inconsistent state. The roll-back re-schedules the updates to run sequentially so restoring – with a temporal delay – the database state.

Compensation is based on a similar philosophy and – on detection of a conflict - initiates a transaction which restores the database to a correct state. In a distributed database environment a transaction may affect tables geographically remote from each other with communications latency. Two-phase commit protocols are designed to ensure consistent state across the distributed database – even if this state no longer corresponds with the real world. Similarly compensating transactions can restore a distributed database to an internally-consistent state which may not represent the real world. Use of such transaction controls typically takes from microseconds to minutes. In an environment when millions of nodes are self-updating locally from sensors (including audio, video) and human input it is clear that conventional database technology does not work. The best we can achieve is local state internally consistent and consistent with (a small slice of) the world of interest, and reconciliation across the distributed database as and when required (i.e. when inconsistent parts of the database are referenced for retrieval or update) using lazy techniques. Alternatively, we re-think the notions of state and transactions.

Data representativity

The problem is to represent the real world inside the database system so that changes caused by events triggered by external factors - humans or machines in the real world - are reflected in the database. The database has to represent values of attributes of entities. However, these attributes can range from simple characters (even then there is the problem of representation in legacy systems while today solved by Unicode) through numeric representations with precision and accuracy to whole text (with language variants) or multimedia objects (with different encodings) or even a binary representation impenetrable without more knowledge of the representation. It is thus necessary to describe the attributes by descriptive metadata, ensure integrity using schema metadata, locate using navigational metadata and restrict usage as appropriate using restrictive metadata.

The structure of the information constructed from the data is a key aspect. Commonly humans structure ‘things’ into hierarchies – for example groups within departments within faculties or schools within a university. However, few universities have such a structure and groups may ‘belong’ to > 1 department, faculties or schools may not exist and research centres may either exist independently of any department or ‘owned’ by several. This would indicate that hierarchies do not represent the real world, and in fact the relationships between hierarchic levels in a data structure have semantics, temporal duration and probability – the latter especially important when dealing with incomplete and inconsistent information.

This should not surprise us; we are all familiar with the non-hierarchic ‘2 parents one child’ graph (which has semantics, temporal duration and probability) and the generalization to many-to-many cardinality. Another example is classification of species and subspecies: simple hierarchies do not match the real world state and again the relationships have semantics, temporal duration and probability. Much of the problem with relational database implementations is because people try to force a real world into a hierarchic representation (using primary and foreign keys) whereas using base tables and ‘linking relations’ allows naturally a n:m relationship (with semantics, temporal and probability attributes) to be expressed. One data model with these properties for the domain of CRIS (Current Research Information Systems) is [CERIF]. In fact this data model has been used in conventional organisational settings as well. As a by-product such an information structure maps directly to hypermedia structures (semantic web and linked open data) such as WWW.

Such a structure has many advantages; for example it is common in some applications for the base tables to be relatively static (append not update) whereas the ‘linking relations’ are frequently updated – indicating rapid changes in inter-relationships of ‘things’. Is this not exactly what is required for the FI? Alternatively, the base tables may be updated frequently whereas the ‘linking relations’ may remain static – examples include detector arrays where streams of data are collected according to a plan – and the plan is encapsulated in the structural relationships between the data streams – at least for a given temporal duration.

Such a representation allows the ‘time machine’ to operate: the state can be recreated at any time slice by retrieval across the ‘linking relations’. How many times do we wish to know the ‘state of the world of interest at a given time or over a given interval? As a side effect storing the temporal information in ‘linking relations’ is much less expensive than storing temporal information with base attributes as in the conventional temporal relational model [Sn94] and allows standard SQL processing without recourse to the temporally-extended TSQL. As a by-product such a representation assists greatly with provenance.

Such a representation allows for probability on the relations between ‘things’ (entities or objects) to be expressed within the ‘linking relations’. This is particularly helpful in many applications where relationships are deduced by humans, inferred by discussion or speculative in a scientific research environment. Of course the ‘intelligence’ of processing the probabilities depends on the capabilities of the system from simple relational calculus through to full fuzzy logic capabilities.

Interoperation – usually to provide a homogeneous view of information to an end-user from heterogeneous sources – clearly requires rich metadata about the attributes and syntax (structure) to be able to match schemas (and more detailed metadata) and to map them to each other. Invariably, additional knowledge processing is required: this may be supplied by humans but progressively more of this task is done by computer systems, using a variety of techniques including graph theory (matching structures) [SkKoBeJe99] or lexical matching with thesauri and knowledge-based reasoning (using domain ontologies related to the attributes and syntax of the domain of discourse) [JeHuKaWiBeMa94]. The current 'half-way-house' of part-machine, part-human reconciliation of databases where only the required resources are integrated and then on a 'pay as you go' basis is the research area commonly referred to as 'dataspaces' [FrHaMa05]. To improve the machine support of this process it is thus necessary to describe the attributes by descriptive metadata, ensure integrity using schema metadata, locate using navigational metadata and restrict usage as appropriate using restrictive metadata – exactly as stated in the first paragraph of this section.

Data quality, veracity and permanence

The purpose of data, especially when structured in context as information, is to represent the world of interest. There are real research issues in ensuring that this is true – especially when the data is incomplete or uncertain, when the data is subject to certain precision, accuracy and associated calibration constraints or when only by knowing its provenance can a user utilise it confidently. Clearly metadata is required to provide the contextual information within which the primary data may be processed. However, there are particular problems in representing incompleteness and uncertainty and in processing such data. Research on modal logics [GaGu03] is providing some hope of solutions.

Validation of data on input is of critical importance to ensure veracity and accuracy. This is best done by validation against constraints stored in the (schema) metadata. However, this is time-consuming and may degrade significantly performance, especially in database update.

Once the data are stored there remains the problem of ensuring they are available and accessible over time. There is a need to preserve digital data from media fading, media migration and loss of ability to process (usually stored in the metadata associated with the dataset or in the associated software). Recently interest in digital preservation has increased, not least due to the cost of collecting (especially scientific) data. A recent report from the EC [Wo10] highlights the opportunities and difficulties with large-scale scientific research data where they are encountered earlier than – but surely will be followed by – commercial data systems. Work is ongoing on standards for digital preservation, technical solutions and business models to sustain digital preservation. A comprehensive resource is available at [APA].

Trust, security and privacy

Security is an issue in any system, and particularly in a distributed system. It becomes even more important if the system is a common marketplace with great heterogeneity of purpose and intent. The security takes the forms:
a) prevention of unauthorised access: this requires identification of the user, authentication of the user, authorisation of the user to access or use a source or resource and provision or denial of that access. The current heterogeneity of authentication and authorisation mechanisms provides many opportunities for deliberate or unwitting security exposure;
b) ensuring availability of the source or resource: this requires techniques such as replication, mirroring and hot or warm failover supported by digital preservation. There are deep research issues in transactions and rollback/recovery and optimisation;
c) ensuring continuity of service: this relates to (b) but includes additional fallback procedures and facilities and there are research issues concerning the optimal (cost-effective) assurance of continuity.

In the case of interrupted communication there is a requirement for synchronisation of the end-user's view of the system between that which is required on the PDA and / or laptop and the servers. There are particular problems with wireless communications because of interception. Encryption of sensitive transmissions is available but there remain research issues concerning security assurance.

The privacy issues concern essentially the tradeoff of personal information provision for intelligent system reaction. There are research issues on the optimal balance for particular end-user requirements. Furthermore, data protection legislation in countries varies and there are research issues concerning the requirement to provide data or to conceal data. A very common security exposure is through 'social engineering' which tricks the end-user into providing information that can be used to compromise a system.

When any end-user purchases online (e.g. a book from www.amazon.com using B2C) there is a trust that the supplier will deliver the goods and that the purchaser's credit card information is valid. This concept requires much extension in the case of contracts for supply of engineered components for assembly into e.g. a car using B2B. The provision of an e-marketplace brings with it the need for e-tendering, e-contracts, e-payments, e-guarantees as well as opportunities to re-engineer the business process for effectiveness and efficiency. This is currently a very hot research topic since it requires the representation in an IT system of artefacts (documents) associated with business transactions and software to reconcile policies so declared.

Management of service levels and quality of service

In the FI the end-user expects services to be delivered with a given quality defined by a service level agreement. The end-user neither knows nor cares how the service is delivered – the underlying data, software, servers, communications systems are virtualised and thus hidden. Thus there is a requirement for service policies offered to be monitored and enforced through recording of parameters against the restrictive metadata including performance levels and costs. There is a need for service level negotiation using the policies in processes related to trust (see above).

Systems design, development, operation and decommissioning

Conventional systems design has typically relied on models and techniques developed in the sixties and seventies. In particular the approaches of Hierarchic Input Process Output from IBM [HIPO], Jackson Structured Programming (JSP) [Ja75] and Jackson Systems Development (JSD) [Ja83]. These data stream led approaches contrasted strongly with the algorithmic approaches of Wirth [Wi71] and Dijkstra [DaDiHo72]. The latter were much better suited to formal proofs using environments such as those of Z, B and VDM although the high cost of formal verification has led to such techniques being restricted to safety-critical applications such as aircraft control. With the arrival of the WWW and the need for additional functionality in the browser and in the Web Services layer, the concept of mobile code was revived and the Java implementation [GoJoStBr05] became dominant with the associated Java Virtual machine platform.

More recently the re-use of software has become topical, analogous to the re-use of artefacts in mechanical or electrical engineering. This has led to the Service-Oriented Architecture Concept [SOA] with variants of static composition (a system developer composes services together to provide a complete application) or dynamic composition (the system environment itself discovers and integrates the services). It is claimed this reduces maintenance and development costs and provides more reliable software. It also circumvents the need for developing new and decommissioning old systems since an evolutionary approach of service replacement using ‘best of breed’ is advocated.

Thus we reach a stage of systems development where we provide services that can be discovered, validated as suitable, composed (dynamically) and executed to meet the requirements of an end-user. Such services require detailed metadata to permit the above operations and there is currently much research in this area. Following on from the work within the GRIDs community and onwards into the CLOUD community, the concept of SOKU (service oriented knowledge utility) has been proposed as a metadata-rich encapsulated serviced which meets the above requirements. SOKUs also have some characteristics found in autonomous agents in that they are loosely coupled to each other and to the operating system environment. One can envisage an ecosystem of autonomous SOKUs interacting dynamically to meet the end-user requirement. One analogy is with a colony of ants or bees; some local autonomy but a commonality of purpose. This links neatly with some approaches for ‘future computing’ which are bio-inspired.

Conclusion

The Future Internet challenges many long-held hypotheses, theories and practice in ICST. The scale and numbers of objects, the heterogeneity and the demands for appropriate security, privacy, trust, permanence and performance all impinge upon traditional ICST engineering practices. However, the challenges are being approached by research both in narrow specialised fields and across a broad front. There is plenty of stimulating research to be done to meet the challenges.

Jan Karel Lenstra

Much of the above benefited from discussions with Jan Karel in the Strategy Task Group of ERCIM. Although his background is more in mathematics and algorithms, he criticised constructively the architectural ideas and informatics challenges presented here. This work is much richer for his contributions as a colleague and friend.

References

[APA] http://www.alliancepermanentaccess.org/

[CERIF] http://eurocris.org/Index.php?page=CERIF2008&t=1

[DaDiHo72] O.-J. Dahl, E.W. Dijkstra, C.A.R. Hoare Structured Programming, Academic Press, London, 1972 ISBN 0-12-200550-3.

[DC] http://dublincore.org/documents/dcmi-terms/

[FrHaMa05] Michael Franklin, Aron Halevy, David Maier: From Database to Dataspaces: A New Abstraction for Information Management SIGMOD Record Vol 34 No 4 December 2005 pp 27-33 available at http://www.cs.washington.edu/homes/alon/files/dataspacesDec05.pdf

[GaGu03] D.M. Gabbay, F. Guenther (Eds): Handbook of Philosophical Logic 2nd Edition. Kluwer Academic Publishers, Dordrecht, Netherlands., ISBN 1-4020-1644-1.

[GoJoStBr05] Gosling, James; Joy, Bill; Steele, Guy; Bracha, Gilad (2005). The Java Language Specification (3rd ed.). Addison-Wesley. ISBN 0-321-24678-0. http://java.sun.com/docs/books/jls/index.html.

[HIPO] IBM Corporation (1974).*HIPO—A Design Aid and Documentation Technique*, Publication Number GC20-1851, IBM Corporation, White Plains, NY, 1974.

[Ja75] M.A. Jackson (1975). *Principles of Program Design*. Academic Press. 1975.

[Ja83] Michael A. Jackson (1983). *System development*, Prentice Hall, Englewood Cliffs, New Jersey, 1983.

[Je99] Jeffery, K.G ‘Knowledge, Information and Data’ September 1999 Paper submitted to Director General of Research Councils proposing the programme that became e-Science; available at http://epubs.cclrc.ac.uk/work-details?w=29021

[Je99a] Jeffery, K G: ‘An Architecture for Grey Literature in a R&D Context’ Proceedings GL'99 (Grey Literature) Conference Washington DC, October 1999.

[Je00] Jeffery, K G: ‘Metadata’: in Brinkkemper, J; Lindencrona, E; Solvberg, A (Eds): ‘Information Systems Engineering’ Springer-Verlag, London 2000. ISBN 1-85233-317-0.

[JeHuKaWiBeMa94] Jeffery, K G; Hutchinson, E K; Kalmus, J R; Wilson, M D; Behrendt, W; Macnee, C A: 'A Model for Heterogeneous Distributed Databases' Proceedings BNCOD12 July 1994; LNCS 826 pp. 221-234 Springer-Verlag 1994.

[SkKoBeJe99] Skoupy, K; Kohoutkova, J; Benesovsky, M; Jeffery, K G: ‘Hypermedata Approach: A Way to Systems Integration’ Proceedings Third East European Conference, ADBIS'99, Maribor, Slovenia, September 13-16, 1999, Published: Institute of Informatics, Faculty of Electrical Engineering and Computer Science, Smetanova 17, IS-2000 Maribor, Slovenia,1999, ISBN 86-435-0285-5, pp. 9-15.

[Sn94] ftp://ftp.cs.arizona.edu/tsql/tsql2/spec.pdf

[SOA] http://www.soa-manifesto.org/

[Tu36] Turing, A.M. (1936). "On Computable Numbers, with an Application to the Entscheidungsproblem". *Proceedings of the London Mathematical Society*. 2 **42**: 230–65. 1937. doi:10.1112/plms/s2-42.1.230. (and Turing, A.M. (1938). "On Computable Numbers, with an Application to the Entscheidungs-problem: A correction". *Proceedings of the London Mathematical Society*. 2 **43**: 544–6. 1937. doi:10.1112/plms/s2-43.6.544.). Reprinted in many collections, e.g. in *The Undecidable* pp. 115–154; available on the web in many places, e.g. at Scribd.

[vonNe45] John von Neumann (1945) First Draft of a Report on the EDVAC; avilable on the web at http://qss.stanford.edu/~godfrey/vonNeumann/vnedvac.pdf

[Wo10] John Wood (chair) Riding the Wave: Final Report from the High Level Expert Group on Scientific Data 2010 available at http://cordis.europa.eu/fp7/ict/e-infrastructure/docs/hlg-sdi-report.pdf

[Wi71] Nicklaus Wirth (1971) Program Development by Stepwise Refinement *Communications of the ACM*, Vol. 14, No. 4, April 1971, pp. 221-227.