ARCHITECTURAL CHALLENGES FOR A DEPENDABLE INFORMATION SOCIETY

Luca Simoncini
Felicita Di Giandomenico
Andrea Bondavalli
Silvano Chiaradonna

Topical day No. 3, World Computer Congress
August 22-26, 2004 — Toulouse, France

INTRODUCTION

Two objectives:

1) **discuss the gap between complexity of envisaged computer-based systems/infrastructures, and the dependability we are today able to justify. How to move towards "ambient dependability".**

2) **provide our views on future directions and architectural challenges to be tackled for approaching an Information Society which we can depend on - as a step towards ambient dependability,**

Ambient Intelligence

Our society is heavily dependent on computerized interconnected systems and services.

- The use of PCs, home appliances, PDAs, wireless phones etc. increases of order of magnitudes the number of untrained users which blindly rely on the myth of "infallibility" of computers.
- EU is stressing the strategic relevance of Europe being the most advanced part of the world based on ICT, by making many sensitive components of our society dependent on computers.
- The statement that European citizens will be able to rely and depend on "Ambient Intelligence" by using dependable computer systems is true only to a very limited extent.
- An ICT-based society needs a very large reorganization.

Istag scenarios

Significant advances have been made in dependability over the past years yet much further research is required considering the future.

In February 2001 the IST Advisory Group published a number of scenarios for Ambient Intelligence (AmI) in 2010.

Here, dependability in general, and privacy and security in particular, emerge as central socio-technical challenges to be addressed.

Devices are ubiquitous, and interact with pervasive networked infrastructures; information flow over open infrastructures, exposed to accidental and deliberate threats.

These scenarios envision new kinds of human relationships and interactions, and powerful technologies to support them. Important perspectives on the dependability of socio-technical systems arise.

A dependable ICT-based society

A dependable ICT-based society will have to cope with:

- Evolution towards more complex services based on legacy systems difficult to integrate.
- Accidental and non-malicious human-made faults, *and* malicious faults of different severity up to the possibility of terrorist attacks.
- A new type of subtle "common mode operational fault" generated by the combination of:
    1) a very large number of computer-controlled systems of common usage, and
    2) a large number of non-trained users operating such systems.

Ambient dependability

Need for a "global" view of the concept of "dependability", which has to start from the basic intrinsic characteristics of the components to grow up and reach reliance in "ambient dependability".

Ambient dependability encompasses not only technological but includes inter and multi disciplinary aspects, (ergonomics, usability, education, sociology, law and government, etc.).

A first step towards ambient dependability is achieving a dependable Information Society, that requires a harmonized effort from a large set of actors.

Challenging points

- Understand and model new threats and new fault types.
- Define methodologies for designing under uncertainty.
- Apply a user-centered approach: design for usability.
- Re-think and redefine the concepts of “architecture” and “system”.
- Architectural frameworks adaptable to functional and nonfunctional properties which provide guarantees on how dependably they are adapting.
- A move towards the definition of extended dependability attributes, like “acceptable availability under attack”.
- New modeling and simulation means and tools for e.g. complex interdependencies, system evolution, evaluation of vulnerabilities related to security.

Where are we today?

In 1995, the Standish Group reported that the average US software project overran

- its budgeted time by 190%,
- its budgeted costs by 222%,
- and delivered only 60% of the planned functionality.

Only 16% of projects were delivered at the estimated time and cost, and 31% of projects were cancelled before delivery, with larger companies performing much worse. Later surveys show improvements, but success rates are still low.

In a UK survey, published in the 2001 Annual Review of the British Computer Society, of more than 500 projects only 3!! met the survey's criteria for success.

In 2002, the annual cost of poor quality software to the US economy was estimated at $60B [NIST, 2002].

Theory and practice

Progress has been made in dependability methods, tools and processes, but still a great gap remains between what is known and what is done.

Current research covers a wide spectrum of critical systems, going from embedded real-time systems, to large open networked architectures (summarized in the CaberNet Network of Excellence http://www.newcastle.research.ec.org/cabernet/research/projects).

However, many industrial engineering designs are still based on best effort processes with limited, if any, application of the known theories showing a relevant educational issue.

Pros and cons of present architectural designs

Most of large-scale infrastructures have been developed connecting stand-alone proprietary systems with ad-hoc solutions

Pros:

- Ad-hoc components make easier system validation
- Limitation of third-party components
- Re-design and updating do not depend on third parties

Cons:

- Components and implementation technologies evolution and obsolescence
- Unflexibility and difficult adaptability
- Needed re-validation for new systems or major revisions

Cons on interaction and interoperability

Systems with slightly different requirements and specs cannot reuse components from previous designs:

- Complete re-design
- Lack of experience from older systems

Interoperability is hard to achieve:

- Different project specifications
  - Different dependability properties
  - Different communication protocols or media
  - …….
- Difficult integration

Trends, new problems and keywords

Increasing number of (maybe non-trained) users:

- New fault types
- New threats (i.e. to privacy and security)

Ubiquity and mobility:

- New threats to security

Evolution, growing complexity, layering of services:

- Vital services and system survivability

Keywords:

- Integration
- Composition
- Recursion:

fault → error → failure

- Usability

Genericity, Openness, Adaptability, Re-use for:

- Design of dependable components and architectures
- Designing architectures for dependability
- Dependable infrastructures from user perspective

Three dimensions

Design of dependable components

Modelling, designing and using **generic, composable, open source, and reusable components** appear very helpful for building systems of systems that can be easily validated and assessed.

Designing architectures/infrastructures for dependability

Another perspective is related to coping with “how to?”.

abstraction, recursion, and incremental verification will definitely help in designing and structuring multi-layers architectures up to the level of complex infrastructures.

Dependable architectures/infrastructures from user perspective

A final perspective is the architectural level that includes the user. **The user is the one who has the final word on system dependability:**

I think the system/service has (optimal/good/sufficient/insufficient) cost/dependability !

Generic, COTS-based architectures for dependable systems

Natural evolution towards more complex services and infrastructures impose an enhancement on how an "architecture" is designed.

Definition, prototyping and partial verification and validation of a generic, dependable, and real-time architecture using COTS and able to avoid the negative points previously listed.

Aim at the definition and construction of an architectural framework such to:

- **reduce the design and development costs.**
- **reduce the number of components used in the several subsystems.**
- **simplify the evolution process of the products and reduce the associated costs.**
- **simplify the validation (and certification) of the products through an incremental approach based on reuse.**

Generic, COTS-based architectures for dependable systems-2

The proposed infrastructure should have the following characteristics:

- Use of generic components (possibly COTS) to be substituted without redesign or revalidation of the system.
- Reliability, availability and safety properties associated to the architectural design and not to intrinsic properties of components.
- Use of a hierarchical approach to ease validation.
- Use of early evaluation methods to support design refinements. An early validation of the concepts and architectural choices saves money and shortens the time to market for a final product.
- Openness of the system that should be able to interface and communicate with other systems through different infrastructures and to adapt itself to the different systems it has to interact with.

(Model-based) dynamic reconfiguration in complex critical systems

In complex networked systems the interdependencies among components are relevant to the QoS provided. The failure of a core node may induce either saturation on other parts of the infrastructure or a cascading effect.

A simplified solution is to pre-plan (off-line) the "best" reaction to system and/or environment conditions, and use it when the condition occurs at run-time. **feasible only in presence of a limited number of well defined situations**

Unpredictability of events requires approaches based on on-line system reconfiguration. To design system architectures able to adapt to run-time changes is very challenging.

Dependability manager

Continuously supervises the system and environment, to identify and apply the appropriate reconfiguration at run-time.

Distributed entities have to be inserted to catch exceptional conditions and to report appropriate signals to the manager.

The definition of the manager includes an evaluation subsystem to provide quantitative comparison among alternatives.

When a system reconfiguration is required, simplified models are solved to devise the most appropriate solution. Different modeling techniques and models solution can be considered and integrated to reach the goal.

A general flexible framework has to be defined, allowing to identify the input parameters of the manager, the metrics of interest and the criteria to base the decision on.

Enhancing methods for dependability evaluation

System evaluation through modeling is very profitable since it supports the prediction on a system before incurring the costs of building it.

New issues are raised by the relevant characteristics of the future systems, that are not satisfactorily dealt with by current modeling methodologies.

The new challenges are connected with the increasing complexity and dynamicity of the systems with implications on system representation and on the underlying model solution.

- State-space explosion
- On-line evaluation
- Integration of experimental and model-based evaluation

State-space explosion and ways to cope with it.

- **strongly limits applicability to large complex systems, or**
- **heavily impacts on the accuracy of the evaluation results.**

Modular and hierarchical approaches identified as directions; however, modularity of the modeling alone cannot be truly effective without a ***modular solution***.

Hierarchical approaches. i) facilitate the construction of models; ii) speed up their solution; iii) favor scalability; iv) help in mastering complexity

Key issues are how to abstract all the relevant information and how to compose the derived abstract models.

Composability, **the ability to select and assemble models of components into a model of the system**:

The overall model is achieved as the integration of small models.

Conclusions and Current Work

Existing gap between complexity of envisaged computer-based systems/infrastructures and the dependability we are today able to justify.

Our view on how to move towards an Information Society which we can depend on - as a step towards "ambient dependability"

- **Generic, COTS-based architectures for dependable systems**
- **Model-based dynamic reconfiguration in complex critical systems**
- **Enhancing methods for dependability evaluation**
  - **State-space explosion and ways to cope with it.**
  - **On-line evaluation as a component mechanism for dynamic architectures**
  - **Integration of experimental and model-based evaluation**