Emergency Control Cockpit

Kemal A. Delic
Hewlett-Packard Co
Grenoble, France
kemal.delic@hp.com

Jeff A. Riley
Hewlett-Packard Co.
Melbourne, Australia
jeff.riley@hp.com

***Abstract*— We live in a world of very large-scale systems which can be described as global, very fast, and often saturated with uncertainty. Talking about global communication and energy supply systems, interconnected financial markets, global business operation platforms or contemporary large social networks, we usually deal with normal operating regimes (~99% of time) while systems might switch into unstable regimes (~1% of time). Very rarely those systems end in emergency state which might degenerate into major catastrophe (huge human, material, environment losses). In this paper we describe the architecture and design details of an emergency control cockpit system, enabling better management of various system crises.**

Emergency, control, cockpit,architecture,design

I. Introduction

We are surrounded, immersed, embedded and engaged with a wide variety of large scale systems which are largely defining our private and professional lives. In the previous century, attention was given mainly to industrial and communication systems which posed **high risk in case of catastrophe**. At that time, we have created the industrial control SCADA system which was especially designed for emergency control [1]. Some important principles are still valid today, despite the fact that contemporary systems are by orders of magnitude bigger, faster and very different – by its very nature. The emergence of hyper-large social systems (e.g. Facebook, Twitter, YouTube) serving more than 1 billion global individuals, has marked the beginning of the 21st century.

Those systems are in a **normal operating regime for 99% of the time**, while perturbances, crises and emergencies are rare, but very important for the system survival [2][3]. Our attention is focused on those rare moments in which the system should be controlled by an emergency system different from the normal regime, control system. Control system theory deals in precise details with all system states, controls and stability, while we will use here only a few practical cases from our experience to outline requirements for the emergency control cockpit – as an embodiment of the emergency control in air traffic domain, as telling example.

Emergency is the critical state of the system caused by the various perturbances, which may evolve into major catastrophe and huge material losses and human lives lost. Timely and accurate taming of any emergency can either avoid those losses or at least reduce them to acceptable level. Anecdotal, well-known examples are airplane crashes, nuclear incidents, space exploration, stock market flashes and deadly spread of viruses (health, IT).

Emergency control cockpit is an embodiment and practical response to such situations widely applicable across many domains. It aims to provide more accurate situation awareness and more timely response and better structured behavior of human controllers. It augments human assessment process, relying on the human cognitive cycles and provides specialized tools to deal with any emergency. Distinctive capability of the cockpit is engagement of the human controller into closed control/management loop, with predefined time constraints and augmented situation and reality awareness.

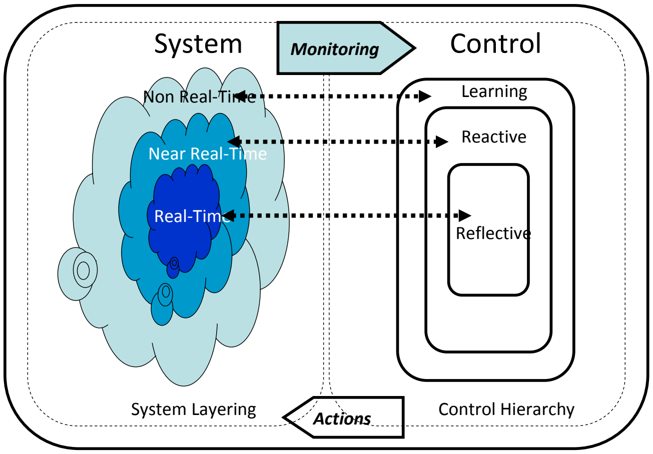
Following our long, practical experiences in various domains we outline architecture and design of an emergency control system which we claim to be effective and efficient in various domains including even control of emerging mega-structure of computing clouds [4].

We introduce the problem of emergency management and follow with layered system architecture with appropriate control component in the second part. In the third part, we provide an example of the very large corporate IT system and introduce key technological design elements. In the fourth section, we outline generic architecture of the emergency control cockpits and describe functional components. **We conclude** with emerging, federated cloud control architecture, and suggest that this conceptual architecture can be applied safely to a wide variety of large scale systems.

In the next section we provide arguments for layered control system and explain design rationales.

II. LAYERED SYSTEM ARCHITECTURE

A system under control can be conveniently divided into three separate layers by the urgency of control reactions: reflective as being immediate or without any latency, then reactive part gives some time to system/human to act in a nearly real-time fashion. The last layer has no real-time constraints and could be described as the learning and memory part of the control system. Layers should be designed differently and closed monitoring-control loop will have appropriate timing constraints. This conceptual architecture (Fig. 1) follows thinking of Sloman [5] and aligns with known PID control paradigm: P-proportional I-Integrative D-Derivative components from the control theory. Important to observe is that these control components are directly corresponding to layering of the system under control. This should be noted here as **the key architecting principle**.



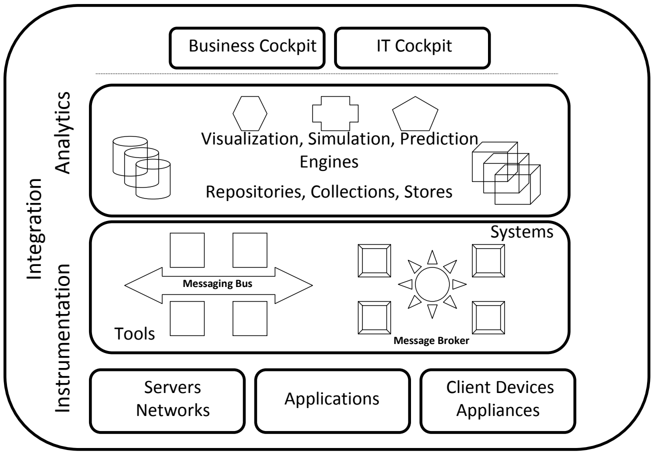
System Control : Conceptual Architecture

Figure 1. Conceptual Architecture of System Control

Proportional control implies immediate adjustment via a feedback loop either amplifying or damping the signal, derivative component enhances/accelerates reaction on the change, and the integrative component provides accumulated/memory effects of the past control signals. Emergency control, however, will address the first layer of control and possibly part of the second, while the third one will act as the memory and learning component. In the next section we apply the same architecture layering approach to the very large corporate IT system and argue that it gives important advantages in case of emergencies.

III. VERY LARGE CORPORATE IT SYSTEM

Large corporations operate through very large IT systems containing tens of thousands of servers, hundreds of thousands of client devices and tens of thousands of applications. The high level architecture shown in Fig. 2 depicts layered, bottom-up instrumentation, integration and analytics systems and repositories. They are feeding into two different types of cockpits: business cockpit used for business management purposes and IT cockpit used for IT management purposes. As they obviously have different focus and operational goals, they are closely inter-related and make Enterprise Management System itself. They fit exactly into the previously indicated architectural layering. The bottom part is all about events and real-time reactions at the instrumentation layer; the middle part is transactional and highly dependable; while the upper part is interactive and analytical enabling management tasks [6].



Enterprise IT Management System

Figure 2. Enterprise Management System

Imagine a situation of rapid **propagation of a virus** within a corporate IT domain. The immediate reaction will be to declare an emergency, suspend suspicious traffic flows and start situation analysis. Once the generic cycle of Detection leading to Diagnosis and finishing in Prognosis is closed, appropriate remedial actions will be executed, and the emergency case data and information will be documented and archived for the future re-use.

In a similar fashion, news feed bringing **breaking news** about important changes in the company market may create emergency conditions in which tools will be activated to arrange add-hoc meetings, assess the market situation and issue appropriate emergency taming procedure(s). In generic terms, emergency taming goes via Situation Assessment leading to creation of the Battle Plan and triggering Prediction Scenarios used to devise the best strategy for ongoing emergency taming.

Another example from our past experience around year 2000 is a **global outsourcing platform** serving 600+ customers for which dependability and security were the key design and architecting criteria [7]. This is by scale, scope and complexity order of magnitude above corporate IT example. It is yet another example of the deployment of emergency control cockpits (Fig. 3) as an effective means to monitor, manage and evolve this real-world, global business platform.

It is rather easy to envision possible problems and consequences for the business in which performances are not closely monitored and managed, and customers isolated. We must ensure that one client's troubles are not propagated across the entire platform, and that a domino effect is avoided by any means and measure. Scale of the platform is global, time constants are measured in minutes and overall value-at-risk is likely huge. Those systems are rightfully called highly-dependable service platforms.

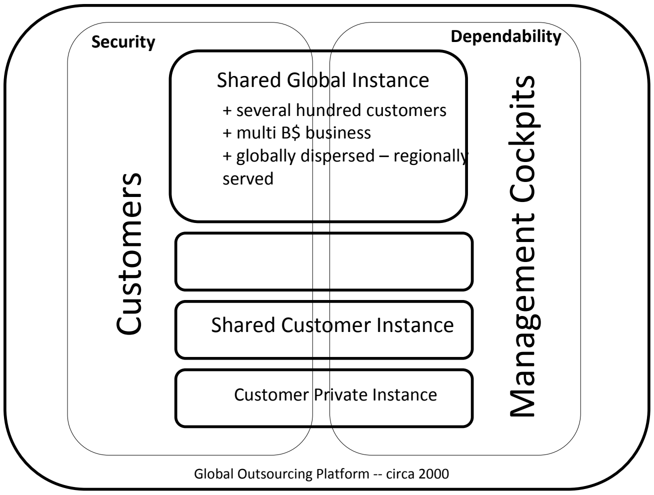

Security
Dependability
Shared Global Instance
+ several hundred customers
+ multi B$ business
+ globally dispersed – regionally served
Customers
Management Cockpits
Shared Customer Instance
Customer Private Instance
Global Outsourcing Platform -- circa 2000

Figure 3. Global Outsourcing Platform

Operating in non-stop fashion, serving global audiences they are under close scrutiny of non-stop media, Wall-Street investors and permanent customer attention. Therefore, outages of such structures are global, breaking-news event with remarkable impact.

IV. CONCEPTUAL ARCHITECTURE OF THE EMERGENCY CONTROL COCKPIT

Critical missions are typically controlled from the large dispatching centers by multiple operators covering different aspects of the flight mission – as in NASA, for example. The same images are seen in the energy and communication management centers and during various military operations. It is always important to take into account human cognitive capacities and mitigate information overload and attention orientation and span. This becomes critical for emergencies, when time is constrained and **inflow of data and information becomes bursty and chaotic.**

We sketch the canvas of the control cockpit layout (Fig. 4), which makes a clear distinction between the normal control system (above) and the emergency control panel (below). Orientation is from the left-to-right with priority on the left – which means that the early warning signals will be fused into groups. Operational parameters are adjacent, so that the operator **can correlate warning signals** with operational parameters.

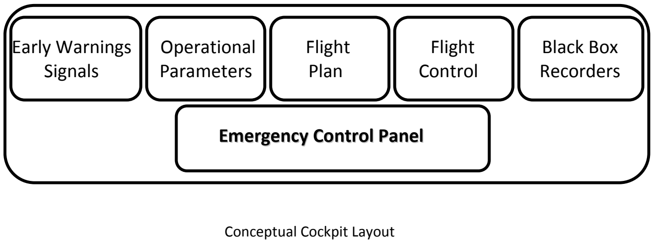


Figure 4. Functional Cockpit Layout

Current position and situation is summarized in the flight plan (middle) so that the current context is clearly given and the future path projected. Flight control parameters are shown on the right-hand side to enable correlation with current state of control. Indications of active recorders are shown on the far-right.

Once an **emergency is sensed, detected,** confirmed and proclaimed, an entirely new panel is lit-up with a very simplified set of indicators and commands based on pre-defined emergency scenarios, and ready for immediate execution. In the next section, we expand further this conceptual schema onto new domains, and give some more technology details of the control cockpit.

II. CONFEDERATED CLOUD CONTROL

Computing Clouds are very large aggregates of servers, applications, network devices and various clients providing new use and business model based on the service fees and not on the ownership of the entire IT fabrics. To give an idea of the clouds scale, the OpenStack framework covers up to 1 million physical, and 60 million virtual, servers under management. Developments have articulated private, hybrid and public clouds while the end state will most likely comprise the federated corporate cloud (Fig. 5), consisting of pieces of the each of above mentioned clouds.

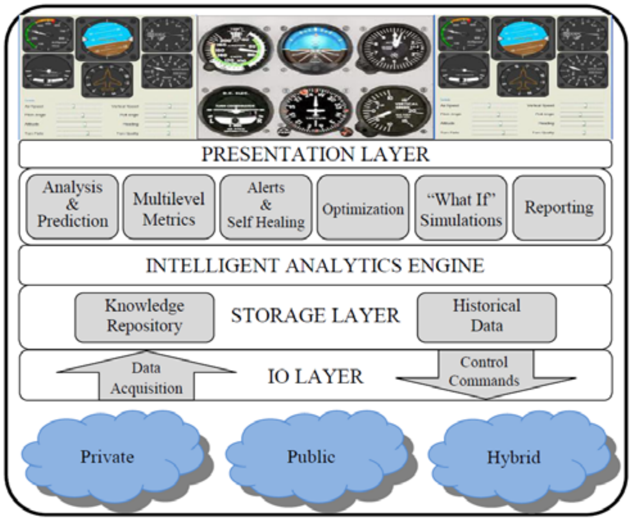


Figure 5. Federated Cloud Control Cockpit

Thus, management of the federated cloud (known as hybrid IT) will become even more intricate and time critical. Here again, we see the necessity of deploying an emergency control cockpit as very valuable as it will enable the safe execution of business tasks on non-owned IT resources.

Private, public and hybrid clouds are providing monitoring feeds going into upper layers with appropriate analytics enabling analysis, prediction, simulation and optimization. Considering the size of clouds, range of operations and criticality of rapid control, traditional methods of using spreadsheets and deploying manual control would be impossible. We argue elsewhere the necessity of creating self-managing IT systems in the future [8]. This will surely enhance capabilities for treating emergencies in an innovative way.

III. EMERGENCY CONTROL FUTURE

The emergency control approach was tried by authors in the 1980s when computing was not at today's advanced level [9], but it was still possible to create working systems and experiment. Even in critical aviation environments [10] AI technologies in advisory roles have been tried and tested at that time. Injection of several new technologies into traditional large-scale systems created new circumstances of

- overall and full global connectivity
- aggregation of the huge number of different systems
- sharp rise in the overall system complexity
- emergence of unpredictable phenomena
- system in a constant state of dynamic equilibrium

leading to unstable regimes and possible emergencies for which we need especially designed system able to deal with such emergencies. A drastic example would be the Fukushima nuclear disaster in 2011, triggered by natural phenomena (earthquake and tsunami), followed by an industrial nuclear disaster which had immediate effect on all stock exchanges worldwide, and significant, long-term (unknown) environmental impact.

In a less dramatic case, rumors spread on the social networks are provoking either mass reaction or very large social movements, so without an emergency strategy in place and without means that the emergency can be controlled, one cannot predict possible outcomes.

Finally, the flash-crash event in 2010 in which an unknown problem in automatic trading provoked losses of $600B in 20 minutes, indicates yet another need for an emergency cockpit approach in which humans will get involved and emergency procedures executed.

In this vein, we believe that the emergency control cockpit concept has an important role to play in the future of large-scale system management. Up until now, emergency control was mainly done via reports and dashboards, while cockpit provides suitable paradigm of the closed loop; enforcing timely appropriate acts of trained cockpit personnel. In the rough analogy to the airplane pilot, dealing with in flight emergency situations routinely via dedicated cockpit and bringing the plane and passengers into safe landing.

ACKNOWLEDGMENT

We thank the reviewers of this paper for providing valuable feedback, and extend our gratitude to all our collaborators on many projects during the last ~ 30 years.

REFERENCES

[1] K. A. Delic, N. Tanaskovic , Designing a Knowledge-Based Emergency Control, Control Engineering, September 1988.p. 268

[2] L. Fisher, Crashes, Crises, and Calamities. Basic Books 2011

[3] A Fake AP Tweet Sinks the Dow for an Instant, Bloomberg News, April 23, 2013

[4] R.Campbell et al, Open Cirrus Cloud Computing Testbed, USENIX, HotCloud'09, June 2009

[5] A. Sloman, Damasio, Descartes, Alarms and Meta Management, Proceedings of IEEE International Conference on Systems, Man and Cybernetics, San Diego, CA (1998) , pp. 2652-2657

[6] Casati F. et al "Enterprise Management Analytics", 11th OpenView OVUA Workshop, Paris, June 2004

[7] K.A. Delic, On Dependability of Of Corporate Grids, ACM Ubiquity Magazine, Vol. 6, Issue 45, Dec 2005

[8] K.A.Delic, J.A. Riley, Architecting Principles for Self-Managing Enterprise IT Systems, ICAS 2007 conference

[9] K.A.Delic, S.V.Tripkovic, A Relational Database Support for Real-Time Advisory Systems, p. 58, Proceedings of IECON'85 Conference, San Franciso, CA Nov. 18-22, 1985

[10] B.L.Belkin, R.F.Stengel, AUTOCREW: A Paradigm for Intelligent Flight Control, in An Introduction to Intelligent and Autonomous Control, Kluwer Academic Publisher, 1993