Fault Management Architecture Assessment

K. Barltrop (JPL/Caltech) and D. Garlan (CMU)

April 11, 2012

Acknowledgements

- Organizer – Kevin Barltrop
- Sub-session chairs – John Day, Dan Dvorak, David Garlan
- Consulting – Dan Dvorak, Lorraine Fesq, Carlos Garcia-Galan, David Garlan, Mitch Ingham, Alex Kadesch, Bob Rasmussen
- Case Study Leads – Mark Brown, Brian O'Hagan, Keith Patrick, Jonathan Rustick, Eric Seale, Judy Tillman

Topics

- Background
- Architecture Assessment
- Assessment Process
- Role of Workshop

Background

- The final report from the 2004 Fault Management Workshop presented a set of 12 Findings and associated Recommendations. While architecture is mentioned in several places, Recommendation 8 focuses specifically on *architecture assessment*:
  - **Recommendation 8:** Assess the appropriateness of the FM architecture with respect to the scale and complexity of the mission and the scope of the autonomy functions to be implemented within the architecture.
- The Workshop report includes a set of opportunities for investment in this area:
  - Capture existing FM architectures and requirements on mature programs. Collect design drivers and implementation decisions in a repository to provide a resource that enables future fault management architects to make better trades.
  - Develop and/or put into practice methodologies for more rigorous architecture specification to enable formal architecture-level analyses and facilitate architecture review and pattern re-use.
  - Develop visual formalisms that facilitate FM architecture design and review, such that the architecture is understandable by system engineers and non-fault management domain experts.
  - Articulate a comprehensive list of functional and non-functional properties for use as figures of merit in assessing FM architectures, and compile a mapping from architectural features to the functional and non-functional properties they promote (including examples of such features).

Response to Workshop Recommendation

- Provide tools and methods for performing a technical assessment that can address three types of questions with respect to fault management:
  - 1. How well does a proposed solution fit a given mission and organization?
    OR
  - 2. How well do other existing solutions fit a given mission and organization?
    OR
  - 3. How well do individual features from existing solutions fit a given mission and organization?

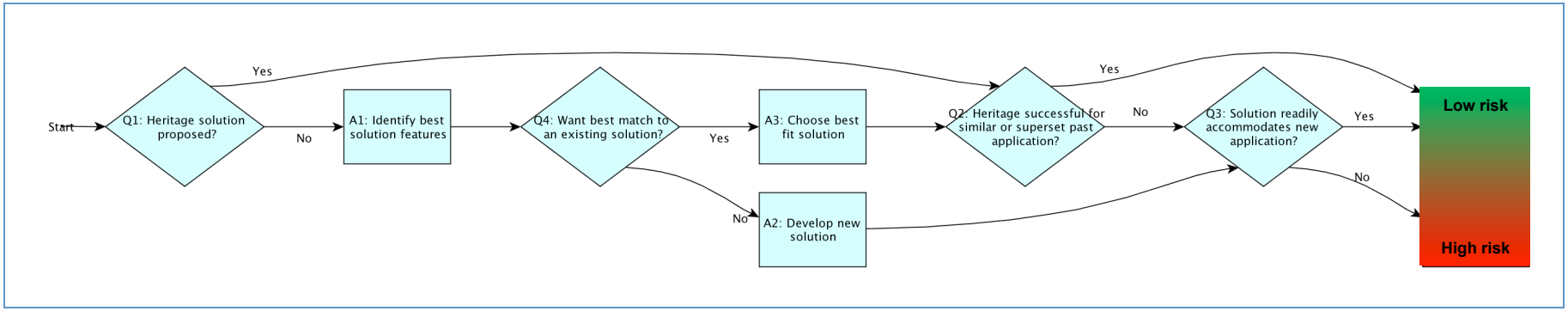
Architecture Assessment

- Idea – Use software architecture notion of "quality attributes" as the basis of an architectural assessment tool and methodology

- What are quality attributes, and how can they be applied to assess fault management architectures?
  - "Software Architecture", presentation by Dr. David Garlan, Carnegie-Mellon University

General Approach

1. Develop a process and structured data resource to support fault management solution trades.
2. Implement an ongoing process and method to collect and maintain the data for past and future projects.
   - NASA FM Workshop serves as pilot for collecting data.
3. Implement a method to allow users to answer any of the three key fault management questions for their individual cases.
   - NASA FM Workshop serves as pilot for demonstrating the use of that method.



Definitions for this Session

- We define *fault management* as the aspects of a mission, such as practices, tools, staff, and on-board hardware and software features, that allow a mission to continue after faults or unexpected events.
- We refer to a *fault management solution* as the chosen combination of practices, tools, and features.
- To understand a particular fault management architecture scenario, we consider:
  - Mission Characteristics
  - Heritage
  - Design Dimensions
  - Implementation Approach
  - Quality Attributes

Role of the Workshop

- The Workshop will serve as a means to:
  - Assess the proposed methodology and selected set of quality attributes through review of historical case studies,
  - Use insights from case study review to make an assessment of a future mission concept in real time, and
  - Provide basis for additional applications to be added to architecture database

- An out-brief will summarize feedback from the participants about the activity

Case Studies

- The following historical case studies have been developed for discussion during the architecture session:
  - Cassini Attitude Control FP, M. Brown (JPL)
  - ISS Autonomous FDIR, B. O'Hagan (JSC)
  - Orion/MPCV, E. Seale (LM-Denver)
  - Chandra, K. Patrick(NGC)
  - SSTI, J. Tillman (NGC)
  - Dawn, J. Rustick (Orbital)
- Discussion will center on the selected quality outcomes for each case study, and an assessment of the quality attributes for that class of application
- The future mission to be assessed will be a crewed mission to a near-Earth asteroid (NEA)

Quality Attributes

- A proposed set of quality attributes have been developed in advance
- As part of the discussion, these attributes will be assessed for:
  - Completeness
  - Applicability (to a given mission type)
  - Level of Abstraction
- Will also develop correlations between quality attributes and mission characteristics, design choices and implementation methods

Analyzability
Appropriateness for Organization
Avoid Unnecessary Interruptions
Conceptual Applicability
Conceptual Integrity
Correctness
Cost For Development
Cost for Development Environment/Tools
Cost for Development Time and Testing
Cost for Operations
Cost For Repeated Work-Arounds
Cost for Training
Degrade Gracefully
Doesn't cause mission loss
Familiarity
Fault Coverage
Integrability
Interoperability
Modifiability during Development
Modifiability during Operations
Modifiability Mission-to-Mission
Modularity
Perceived Cost/Benefit
Preserve Resources and Opportunities
Reduce Recovery Time
Reliability
Reusability
Safety
Scalability
Testability
Thrustworthiness
Tolerate Modeling Errors
Usability/Operability

Session Logistics

- 10:30 – Full session in Bonnet Carre
- 11-12:30 – Split into 3 sub-sessions, each discusses first case study
- 12:30-1:15 – Lunch
- 1:15-2:45 – Split into 3 sub-sessions, each discusses second case study
- 2:45-3:00 – Break
- 3:00-3:45 – NEA mission presentation
- 3:45-5:00 – Split back into sub-groups to apply assessment methodology to NEA mission

ARCHITECTURE ASSESSMENT SESSION INTRODUCTION

Topics

- More logistics
- More Specifics on Architecture Assessment
- Assessment Process
  - General Approach
  - Process Overview
  - FM Architectural Assessment Database

Session Logistics, Revisited

- 10:30 – Full session in Bonnet Carre
- 11-12:30 – Split into 3 sub-sessions, each discusses first case study
  - A: ISS FDIR, Chandra (D. Garlan) [Bonnet Carre]
  - B: Orion, Cassini AACS (D. Dvorak) [Queen Anne Parlor]
  - B: Dawn, SSTI/Lewis (J. Day) [Ursaline Salon]
- 12:30-1:15 – Lunch
- 1:15-2:45 – Discuss second case study
- 2:45-3:00 – Break
- 3:00-3:45 – NEA mission presentation [Bonnet Carre]
- 3:45-5:00 – Split back into sub-groups to apply assessment methodology to NEA mission

Architecture Assessment

- Additional details on assessing architectures [D. Garlan]

Historical Case Studies

- Each case study discussion will follow the same basic outline:
  - Case study introduction and background
  - Description of mission characteristics and how they are captured
  - Description of design and implementation characteristics and how they are captured
  - Description of quality outcomes – explain context and engage group to leverage their experiences

- Discussion should center on :
  - Assessment of proposed quality attributes
    - Completeness
    - Applicability (to a given mission type)
    - Level of Abstraction
  - The selected quality outcomes for each case study, and the mission and design/implementation characteristics that affect these outcomes
  - Prioritization of quality attributes for that class of application (e.g., mission type)

Application to Future Mission

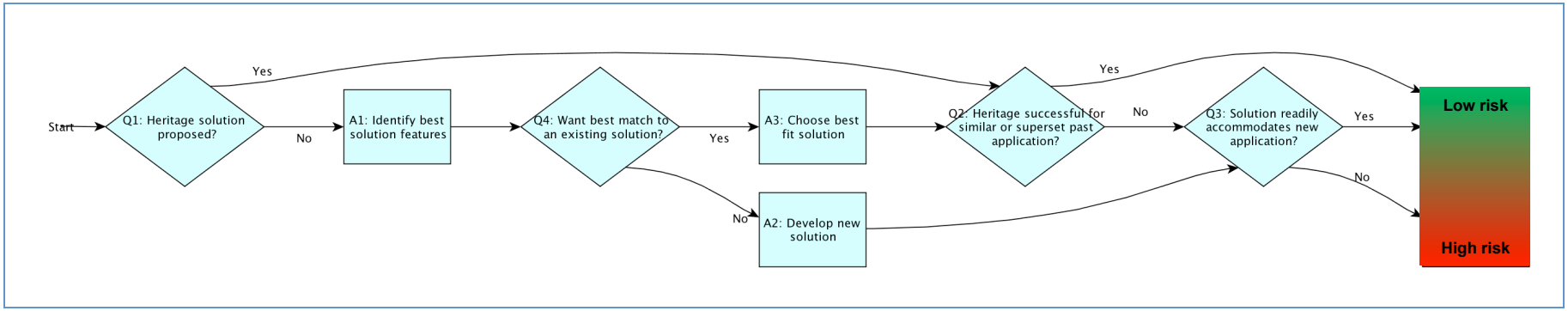
- Description of NEA mission will be presented to full session via WebEx
  - Victoria Friedensen/Dan Mazanek
- After presentation, will split up into sub-sessions to apply assessment approach to NEA mission. Sub-sessions should identify:
  - Significant/important quality attributes for this class of mission
  - Related design and implementation approaches that support identified quality attributes
- Integration of discussion results by sub-session chairs, and presented at report-out on Day 3
  - Option: could arrange joint discussion with full session at 4:30

Assessment Process Overview

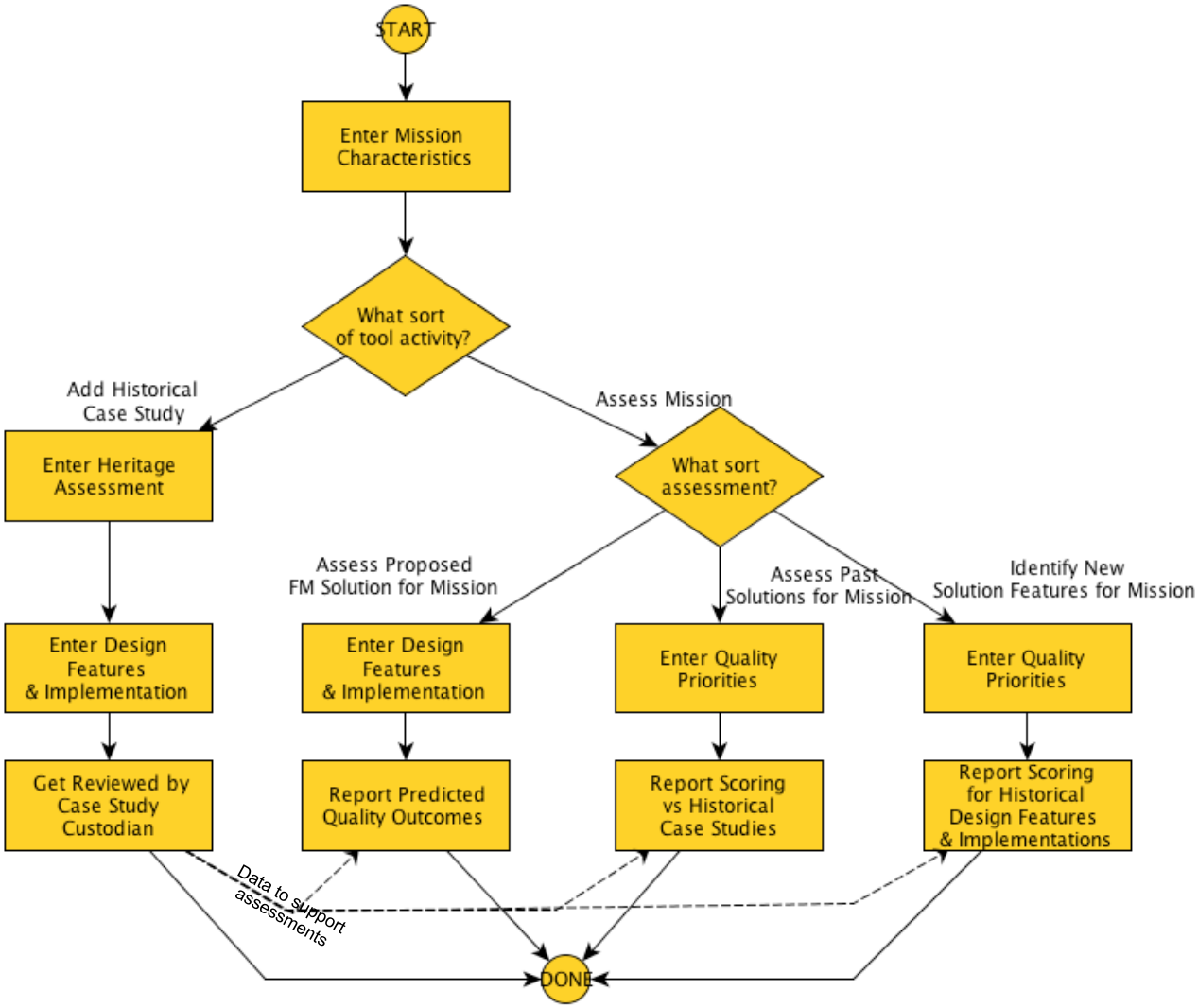
- The assessment process consists of two key elements:
  1. A top level process flow for examining the heritage risk story.
  2. An online database and reporting tool to ground the assessment in measureable data.

Heritage Risk Assessment Process

- We begin with a heritage risk assessment covering at least these areas of a fault management solution:
  - Staff -- Engineering practices
  - Analyses & design tools -- Flight software
  - Flight hardware -- Mission design
- The figure below depicts the assessment flow.
  - Note that even a difficult-to-use solution, can be applied successfully to identical missions once it has been debugged sufficiently.
  - A project may also iterate this process across multiple aspects of the architecture and across multiple changes to the architectural approach.
  - Details for each box are now explained…



Flow for using the Database



Mission Characteristics

- Mission features significantly affect the how well certain designs, tools, and practices will work.

	APPLICATION CATEGORY	
COMPLEXITY	Diverse Activities	Major System Configurations
		Operating Modes
	Dynamic Environment	Mission Phase Environments
		Environmental Variation
	Configuration Complexity	System Interactions
		Cross Strapping and Redundancy
	Critical Timing Windows	Performance Windows
		Health and Safety Windows
	Large Comm Lag	Line of Sight Propagation Delay
		Outage Delays
		Network Propagation Delays
CRITICALITY	Risk Tolerance	Flight Crew Safety
		Ground Bystander Safety
		System Safety
	Investment	Stand Alone Investment
		Infrastructure Investment
	Unique Opportunities	Science Opportunities
		Prestige Opportunities

Design Dimensions

- Fault management largely evolved out of ad-hoc solutions to the question: *What should we do when something goes wrong?*
- An examination of fault management across domains and implementation approaches reveals recurring dimensions of designs.
- Often we are unaware of these because they are not explicitly called out in the design.

DESIGN	
NAME	DIMENSION
Knowledge	
	Representing
	Estimation
Assessment	
	Desired States
	Discrepancies
	Discrepancy Tolerance
Response	
	Strategy
	Constraint Checking
	Coordination
	Influence
	Mitigation
	Character Control
	Synchronization Granulari
	Synchronization Control
	Priority Accommodation
Operations	
	Visibility
	Modification

Implementation Approaches

- Organizations introduce numerous implementation constraints that are often driven by cost phasing, trustworthiness, and history.
- This often requires an organization "pick its poison" when choosing an approach.

MEANS	DEPLOYMENT	ORGANIZATION	ORGANIZATION DRIVER	PARTITIONING	THREADS	STANDARDIZATION	DESIGN SPECIFICATION

Quality Attributes

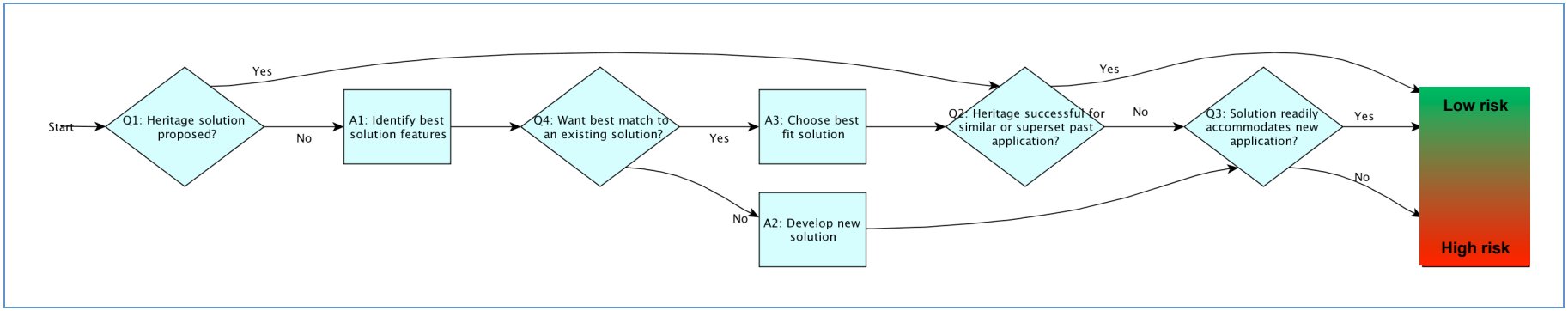
- Organizations have begun looking beyond the immediate requirements of a project, to consider other attributes that greatly affect the outcome of a project.
- We can get stuck with unpleasant results from heritage solutions where these attributes were not considered.

Analyzability
Appropriateness for Organization
Avoid Unnecessary Interruptions
Conceptual Applicability
Conceptual Integrity
Correctness
Cost For Development
Cost for Development Environment/Tools
Cost for Development Time and Testing
Cost for Operations
Cost For Repeated Work-Arounds
Cost for Training
Degrade Gracefully
Doesn't cause mission loss
Familiarity
Fault Coverage
Integrability
Interoperability
Modifiability during Development
Modifiability during Operations
Modifiability Mission-to-Mission
Modularity
Perceived Cost/Benefit
Preserve Resources and Opportunities
Reduce Recovery Time
Reliability
Reusability
Safety
Scalability
Testability
Thrustworthiness
Tolerate Modeling Errors
Usability/Operability

ASSESSMENT PROCESS DETAILS

Heritage Risk Assessment Process

- We begin with a heritage risk assessment covering at least these areas of a fault management solution:
  - Staff -- Engineering practices
  - Analyses & design tools -- Flight software
  - Flight hardware -- Mission design
- The figure below depicts the assessment flow.
  - Note that even a difficult-to-use solution, can be applied successfully to identical missions once it has been debugged sufficiently.
  - A project may also iterate this process across multiple aspects of the architecture and across multiple changes to the architectural approach.
  - Details for each box are now explained…



Q1: Has a heritage solution been proposed for a new mission?

- Heritage should be considered in each of these areas:
  - Hardware
  - Staff
  - Practices
  - Software
  - Tools
  - Mission Design
- Heritage should be examined for multiple aspects of the entire fault management solution, such as high level software framework, system redundancy, local fault handling, etc…
- Breaking of heritage in even one area, such as by the introduction of new staff or tools, can introduce risk, especially if the consequences of change are not adequately identified and mitigated.

Q2: Has this heritage been successfully used for past similar or enveloping applications?

- Points to consider:
  - Did past application use the same hardware, tools, people, software, mission features?
  - Did past application avoid cost and schedule overruns?
  - Did past application avoid near-miss situations related to design flaws?
  - Is it possible that the past applications got lucky in avoiding certain pitfalls?

Q3: Does the proposed solution readily accommodate new applications?

- Points to consider:
  - How well has solution been adapted for new applications in the past?
  - Was the solution deliberately developed to support easy and reliable adaptation?

Q3, Cont'd

- **Observation**: The solution that ultimately works is the one that provides "sufficient" correspondence between the things being managed and the solution's representation of those things.
  - You know you have sufficient correspondence when you have a system that works correctly.
- So how easily does a given solution allow one to achieve that "sufficient" correspondence?
  - The big challenge comes from determining what aspects (states, constraints, objectives, relationships) of the system and the world to represent and with what degree of fidelity.
  - As a matter of practice, we make choices about that correspondence by any of several methods such as trial and error from testing, by rules of thumb, by organizing states and modes for the system, and/or by modeling the physics of the system.

Q3, cont'd

- How well does the solution allow the operator to implement a design in terms of the specific concepts of fault management?
  - Does the development environment provide useful references tied to fault management, such as the notion of errors, faults, and responses?

Q4: Do we need a proposed best matching heritage solution for a new mission?

- Points to consider:
  - Which heritage solution has done well for similar missions?
  - Given the mission attributes, which solution best fulfills the quality priorities of the new mission?
  - We can filter and rank data from past missions to illustrate that matching.

Q5: Are we looking for a new fault management solution?

- Points to consider:
  - What solution techniques and features (practices, system design, and tools) best fulfill the quality priorities for the mission?
  - What architectural solution provides that set of techniques and features?