Example: e Claims Exchange

Requirements highlights:

- automatically process insurance claims: decide whether to allow or deny, calculate payment, transfer funds
- support multiple lines of insurance: dental, drug, vision, general health, etc. (even home and auto, if feasible)
- support all large insurance companies
- support group or individual insurance
- insurers maintain up-to-date coverage information and can restructure insurance plans
- enroll millions of persons
- support submission of claims via the Web
- response time under 3 seconds
Goals, Principles and Constraints

- Data producers (enrollment, insurance plan editor, etc.) and consumers (adjudication) must not be designed separately, or they will not work together smoothly. Data schema must be specified before all else.

- To achieve required extensibility, insurance rules must be externalized (stored as data).

- Security requirements (access control) can be fulfilled by re-using adjudication functionality (security principals and controlled resources are data entities, and rules can be attached to them already).
Writing the Software Architecture Document

1 Introduction

1.1 Motivation and intent

1.2 References to other documents

1.3 Important business considerations

**Hint:** The introduction should be brief but thorough. All requirements and other specifications or sources of input must be referenced. The section on business considerations should be given some attention.
2 Technical requirements

- Platforms
- Portability
- Hardware available
- Existing systems

**Hint:** This section details the decisions already made that you are required to work with (e.g. legacy systems, platforms to be used, portability requirements).
3 Application architecture

**Hint:** Mention user-facing applications and elaborate user interface architecture.
Writing the S.A.D. (Continued ...)

4 Architectural paradigm

**Hint:** Explain the general idea of the system and try to fit the problem into a well-known architectural paradigm. Explain the reason why you choose some particular paradigm.
5 Data architecture

**Hint:** Explain how to load, store, write, and warehouse the system’s data. Address issues of the physical mechanisms (RDBMS vs. OODB vs. some other choice - specify vendor and/or product if appropriate, security, middle-ware, etc.)
Writing the S.A.D. (Continued ...)

6 Run-time and physical architecture

**Hint:** Explain how processes/threads communicate. Clearly describe allocation to hardware (e.g. by using deployment diagrams.)
7 Technology choices

▷ Databases
▷ Languages
▷ Application servers
▷ Web servers
▷ Libraries
▷ Networks and Inter-Process Communication (IPC)
8 Module architecture

▷ Source code
▷ Reuse strategy