

npro ee n s n or Con eren eon nte te Des n n ro ess e no o D Austn es De e er 4
e not rest tot ete e e ro ess ro apure ents to ste Ar teture n n

NT

D
D

D

T

T

C

T

A

T

A

A A

oftware research Lab

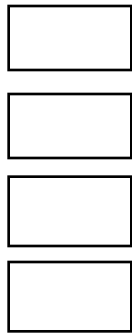
the developer regards the I... agent as an ally in the effort to produce high quality software then the conflict can be avoided. If they regard the I... agent as an enemy put there to hinder them, then the conflict will become central to their relationship.

INT

C B D C

During the... software can be to pay an important role in enabling... AAs development... and more complex spacecraft... Unfortunately... AAs culture of prototype and test... not carry over well to the software... A number of factors affect software... significant... different... different... B... states the complexity of software in relation to its... software as no duplicate components... the accuracy of abstraction... software... pure... a... and... behavior... changes in behavior in response to... changes in input. In addition to these software is seen as more capable than hardware because of the accuracy of abstraction... most engineers believe that it is easier to alter software than hardware in response to changing requirements.

Although software was not... in the... accident... the subsequent inquiry... created a chance to assess... aspects of... development processes... the... commission... of... independent... of... development processes as a significant factor in the... accident... there was no process... or... what... problems that arose in the... engineering processes and... in particular... independent... assessment... were accepted... in the... of... safety... pressures... the role of... separate safety panels was reduced... two subsequent... reports warned that software is underrepresented in... safety... and that... any of the... states that contributed to the... accident are now being repeated... with respect to software... these reports recommended that... AAs adopt software... or... or a... future



approach be adopted, or a such requirements

Having produced a clearer representation the I team then proceeded to identify properties of the specification that should be consistent and complete. A consistency property is that there should be no combination of conditions or where two different failure recovery actions are specified. A completeness property is that every possible combination of failure conditions should have some recovery action specified. These properties were tested by converting the tabular representations into a formal logic. In this case C and using a tool to test for these properties. A significant number of consistency errors were found, there were combinations of conditions or where more than one recovery action was specified. These were traced to a problem with the ordering of the requirements. The correct functioning of the FDI software depends on the tests described in these requirements being carried out in the order that the requirements are given. However this is not stated explicitly. As significant contrast an earlier and a observation by the I team.

At this point the I team would have gone on to check the validity of the requirements against a failure modes and effects analysis of the bus architecture. However at this point in the case study the I team found out that this section of the requirements was being substantially rewritten. Hence they delayed further analysis until the new version became available.

This case study illustrates two interesting points about the work of an I team. First the I analysts often create their own representation or example of a portion of the developer's specification. However these alternative representations are never given to the developer to use in place of the original specifications. This is to ward against the danger of the I team being drawn into developer work and possibly losing their independence on subsequent analyses of these components.

Second the I team have their own discretion on how much analysis to perform. For example the analysis does not stop when the first error is encountered. It is an obvious risk to the error at this sense of the I team to assume that a risk will be able to proceed with other types of analysis. However there is a point beyond which further analysis adds little extra value, or example when a number of errors are encountered, or when as in this case a rewrite is underway.

D - - C - - T

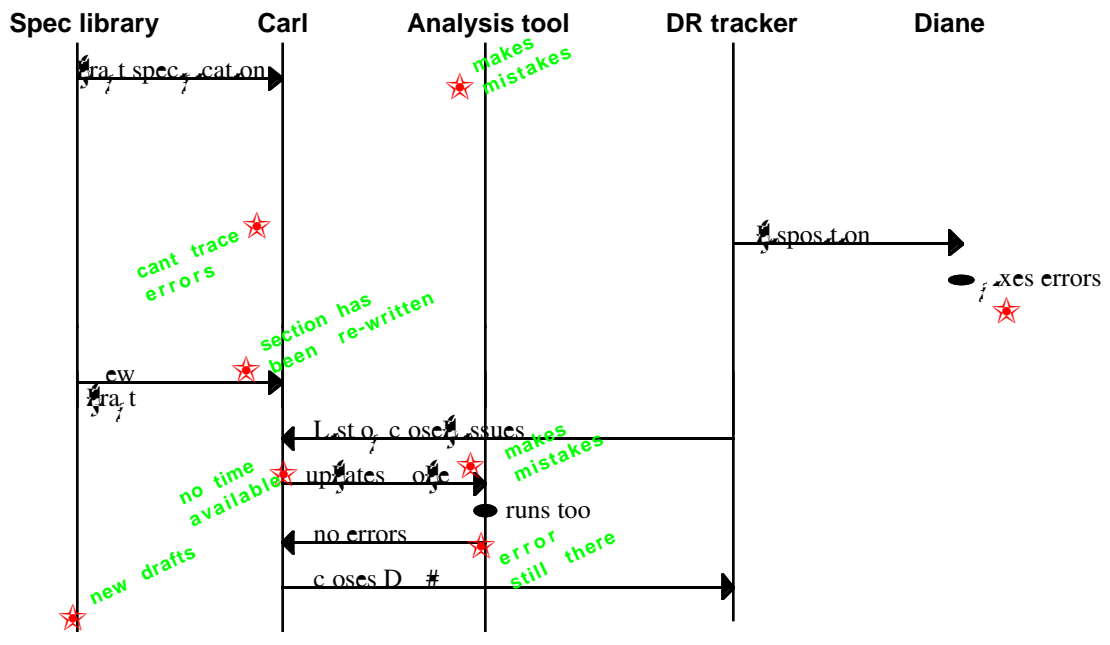
Having described the basic I process and illustrated it with a case study we now discuss some of the difficulties faced by I in carrying out their role in current research. As investigators these problems and see in ways of overcoming them.

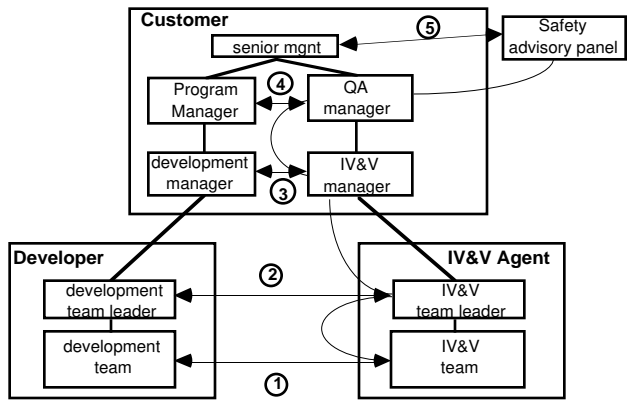
The following difficulties are inherent in the relationships between the developer, customer and I agent. One of these arises as a direct result of the contact roles of I and developer, others are to do with resource pressures and the need for timely results.

- 1. Scope of work** A complete meta-level analysis of the entire system is necessary. Effort has to be allocated so as to maximize effectiveness. For example a criticality analysis might be performed to determine which components need the most scrutiny. As a solo actor, effort needs to be allocated at the right points in the development of a product or a document so that the product is mature enough to be analysed, but not so mature that it cannot be changed.
- 2. Goal of the project** To be most effective I reports are needed as quickly as possible. There is always a trade off between the delivery of an interim product to the I team and the completion of an analysis of that product. During this time the developer process continues. Hence the I analyst takes too long in the results that be available too late to be useful. In general the earlier an error is reported, the cheaper it is to correct and the less reluctant the developer is to fix it.
- 3. Contact between the developer and the I team** Contact between the developer and the I team is crucial to analysis. The I team needs to maintain independence whilst ensure they obtain enough information from the developer to do the job. From the developer's point of view, interaction with the I team represents a cost overhead which can interfere with project deadlines. Inevitably the I contractor has less access to the developer team than a solo.
- 4. Information flow** Documentation for the developer team is usually available to the I contractor in draft form to facilitate early analysis. The drawback is that documents may be revised while the I team is analysing them. Inevitably the results of the analysis are relevant before it is finished.
- 5. Power relations** The I contractor is by necessity constrained by discretion over the types of analysis to perform on different products and which problems to report. It is vital to the effective use of I that the I contractor prioritizes the problems that are reported. In top priority problems are reported, this may swamp the communication channels with the developer and the customer and compromise the credibility of the I agent.
- 6. Dispute resolution** The I contractor may have jurisdictional effectiveness across especially the developer and customer. If a dispute arises, the I contractor is often problems caused by I have cost and security implications and in such circumstances the customer may be unwilling to listen to the effectiveness of I. Then depends on having a credible advocate within the customer or organization.

Conclusion on project

In order to investigate these problems further we have developed a set of scenarios describing particular activities and used these to explore where coordination problems occur. Easterbrook describes these scenarios





As paper has examined the role of I in the software development process, concentrating especially on its role in requirements and design processes. I provides an independent assessment of both development and operational risks. It helps to identify safety, reliability and performance concerns early in the software lifecycle and has generally been demonstrated to save money throughout the development of errors.

The role of I is complementary to that of A. Where A focuses on checking that appropriate standards and process models are applied, I focuses on the technical integrity of the software through analysis of specifications, designs, code and other documentation. Hence I will ensure that the requirements are complete and that a proposed system architecture will meet the requirements and that traceability is demonstrated against requirements, designs and test cases.

An interesting property of the I process is that the I agent can play a role as a process improver agent, for a number of reasons. First, the recommendations made by I in response to errors often address ways to prevent similar errors occurring in the future. Second, the I team have so far exhibited a willingness to apply new techniques and tools, especially where these put perceived gaps in the analysis process, by the developer. These new techniques and tools demonstrate their value in identifying errors. The developer team may choose to adopt the techniques. Finally, the presence of an I contractor provides an incentive for the developers to improve their own internal

