

Appears in Proceedings, Third IEEE International Symposium on Requirements Engineering (RE'97), January 5-8th, 1997, Annapolis, Maryland, USA.

Formal Methods for V&V of partial specifications: An experience report

Steve Easterbrook and John Callahan
{steve,callahan}@cs.wvu.edu
NASA/West Virginia University Software IV&V Facility
100 University Drive
Fairmont, WV 26554

Abstract

This paper describes our work exploring the suitability of formal specification methods for independent verification and validation (IV&V) of

ment teams.

In section 2,

IV&V contractor has less access to the development team than is ideal.

C&C MDM acting as the bus controller
Detection of transaction errors in two consecutive processing frames
errors are on selected messages
the RT's 1553 FDIR is not inhibited
A backup BC is available
The BC has been switched in the last 20 seconds
The SPD card reset capability is inhibited
The SPD card has been reset in the last 10 major (10 second) frames
The transaction errors are from multiple RTs
The current channel has been reset within the last major frame

OR

T	T	T	T
T	T	T	T
T	T	T	T
T	T	T	T
T	T	T	T
T	T	T	T
T	T	.	.
.	.	T	T
T	T	T	T
T	F	T	F

A
N
D

tant in tracing problems back to the informal specification, and in convincing the development team that there really is a problem.

The first step was to produce an SCR model of the specified FDIR behavior. At this stage we

Current Mode	Conditions											Next Mode
	errors in two cons. frames	bus swch'd last frame	bus switch inhibit	bus swch'd this frame	backup BC avail.	BC swch'd in last 20 sec	card reset inhibit	card reset last 10 frames	errors from mult. RTs	channel reset last frame	channel reset inhibit	
Normal	@T	-	-	F	-	-	-	-	-	-	-	switch buses
	@T	-	T	F	-	-	-	-	-	-	F	reset the channel
	@T	T	-	F	-	-	-	-	-	-	F	reset the channel
	@T	-	-	-	-	-	F	F	T	T	-	reset the card
	@T	-	-	-	-	-	F	F	T	F	T	reset the card
	@T	T	-	-	-	-	-	-	F	T	-	switch RT to backup
	@T	F	T	-	-	-	-	-	F	T	-	switch RT to backup
	@T	T	-	-	-	-	-	-	F	F	T	switch RT to backup
	@T	F	T	-	-	-	-	-	F	F	T	switch RT to backup
	@T	-	-	-	T	F	T	-	T	T	-	switch BC to backup
	@T	-	-	-	T	F	T	-	T	F	T	switch BC to backup
	@T	-	-	-	T	F	-	T	T	T	-	switch BC to backup
	@T	-	-	-	T	F	-	T	T	F	T	switch BC to backup
	@T	-	-	-	T	T	T	-	T	F	T	switch all RTs
	@T	-	-	-	T	T	T	-	T	F	T	switch all RTs
	@T	-	-	-	T	T	-	T	T	T	-	switch all RTs
	@T	-	-	-	T	T	-	T	T	F	T	switch all RTs

Table 2: An SCR Mode transition table. Each of the central columns represents a condition, showing whether it should be true or false; ‘-’ means “don’t care”; ‘@T’ indicates a trigger condition for the mode transition. The four columns of table 1 correspond to the last four rows of this table. The semantics of SCR require this table to represent a function, so that the disjunction of all the rows covers all possible conditions (coverage), and the conjunction of any two rows is false (disjointness).

that it leads to, but also for the removal of ambiguities and for improved understanding. For this benefit, it is the *process* of formalization, rather than the end product that is important.

The fidelity problem is really a special case of a more general problem: management of consistency between partial specifications expressed in different notations. For instance, the AND/OR tables have a clear relationship with the SCR mode tables, but if we make a correction to one of the AND/OR tables, it is fairly tedious to identify the corresponding correction in the SCR tables. Similarly, each time the developers of a formal specification (e.g., a state transition table) make a change, it is often difficult to identify the corresponding change in the SCR tables. This is a problem that is common to all formal specifications, and it is a problem that is often overlooked. The SCR mode tables provide a way to manage consistency between partial specifications, and they provide a way to manage consistency between partial specifications expressed in different notations.

t e assorted partial specifications drawn from differ-
ent

continuing the experiments described in this paper by examining how model checking can be used to validate the specifications.

Acknowledgments

Our thanks are due to Chuck Neppach and Dan McCaugherty for many interesting discussions of the work presented here, and to Frank Schneider, Edward Addy, John Hinkle, George Sabolish, Todd Montgomery and Butch Neal for detailed comments on earlier drafts of this paper. This work is supported by NASA Cooperative Research Agreement NCCW-0040.

References

- [1] V. Basili. The experience factory and its relationship to other improvement paradigms. In *Proceedings of the 4th European Software Engineering Conference, Garmish-Partenkirchen, Germany, September 1993*.
- [2] J. Calla an and T. Montgomery. An approach to verification and validation of a reliable multicast protocol. In *Proceedings of the ACM International Symposium on Software Testing and Analysis (ISSTA)*, January 1996.
- [3] D. Craigen, S. L. Gerhart, and T. Ralston. Formal methods reality check: