

	2017 International Conference on Software and e-Business (ICSEB 2017), Hong Kong, 28-30 December, 2017.
	Paraconsistent model checking: Logics, translations and examples
	○ <u>上出哲広</u> (Norihiro Kamide)
	<p>Model checking is a formal and automated technique for verifying concurrent systems. Paraconsistent model checking, which is also called inconsistency-tolerant model checking, is a model checking paradigm that can appropriately verify systems with inconsistencies. Logics, translations and examples for paraconsistent model checking are presented in this talk. Paraconsistent linear-time temporal logic (pLTL) and paraconsistent computation-tree logic (pCTL), which can suitably represent inconsistency-tolerant reasoning, are introduced. These new logics are natural extensions of the standard temporal logics: linear-time temporal logic (LTL) and computation-tree logic (CTL), typically used in model checking. Translations from pLTL and pCTL into LTL and CTL, respectively, are defined, and theorems for embedding pLTL and pCTL into LTL and CTL, respectively, are proved using these translations. These embedding theorems allow us to reuse the standard LTL- and CTL-based model checking algorithms to verify inconsistent systems that are modeled and described by pLTL and pCTL. Some illustrative examples for paraconsistent model checking are also presented on the basis of the proposed logics and translations.</p>