

CHESS to Mobius (SAN) model transformation TODO List:

- Lack of **dependency control** when compiling generated models in Mobius. Saving the models immediately after they have been created and imposing sensible dependency rules, would fix this issue. As a workaround, manually compile models in Mobius to ensure dependencies.
 - not applicable, it's a Mobius issue and we have already a workaround
- **Reward** in Mobius: extend the transformations to generate a reward model in Mobius. The CHESS model should have a Boolean expression (AnalysisContext) to specify the system failure. This serves as the basis for the Reward model generation. The expression can be specified already ("measure" property of a <<CyberSecurityAnalysis>> UML Class) the transformation must be extended.
 - mandatory; as fundamental support, a SAN model shall be generated starting from the Boolean expression provided through the AnalysisContext. We need an example of SAN model for the reward and to define the set of transformation accordingly.
 - optional (low priority): call explicitly Mobius Analysis from CHESS via script (a la Deem) and propagate results back to CHESS (report). Probably City knows how to use command line Mobius.
- **Cleansing** activities: extend the transformations. Cleansing can be modelled in CHESS as a Transition of an Error Model State Machine, stereotyped as <<Repair>> and connecting a <<Degraded>> state to the <<Normal>> state.
 - optional : We need an example of SAN model with the cleansing and to define the set of transformation accordingly.
- **Response Time**: we have the possibility to model it in CHESS. Not implemented in the transformations.
 - optional (would allow multi-concerns analysis): We need an example of SAN model and to define the set of transformation accordingly.
- Define condition of "common failure mode" for replicated components in Mobius. Transformation should be updated accordingly
 - To be defined by City

Created: 5 June 2019

Last updated: 5 June 2019