Appendix C

Formal Description of Scenario 1

section *Framework* **parents** *standard_toolkit*

$[CHAR, DATE, DECISION]$
$LIGAND == \mathrm{seq}\, CHAR$
$RECEPTOR == \mathrm{seq}\, CHAR$
$CONFIG == \mathrm{seq}\, CHAR$
$RESULT == \mathrm{seq}\, CHAR$
$LIGANDS == \mathbb{P}\, LIGAND$
$RECEPTORS == \mathbb{P}\, RECEPTOR$
$RESULTS == \mathbb{P}\, RESULT$
$DEEP\_ALIGN\_RESULT == \mathrm{seq}\, CHAR$
$YES\_NO ::= yes \mid no$
$USER\_INPUT == \mathrm{seq}\, CHAR$
$PREVIOUS\_RESULT == (LIGAND \times RECEPTOR \times CONFIG \times DATE) \mapsto RESULT$
$PREVIOUS\_RESULTS == \{PREVIOUS\_RESULT\}$

$$dockingWithConfig : (LIGAND \times RECEPTOR \times CONFIG) \rightarrowtail RESULT$$

$$\forall l : LIGAND;\ r : RECEPTOR \mid l \neq \varnothing \land r \neq \varnothing \bullet \exists c : CONFIG;\ res : RESULT \mid c \neq \varnothing \bullet dockingWithConfig(l, r, c) = res$$

Docking_AutoDockVina

$ligand? : LIGAND$

$receptor? : RECEPTOR$

$config? : CONFIG$

$result! : RESULT$

$config? \neq \varnothing \land result! = dockingWithConfig(ligand?, receptor?, config?)$

MolecularDockingEnvironment_Raccoon2

$ligands? : LIGANDS$

$receptors? : RECEPTORS$

$config? : CONFIG$

$results! : RESULTS$

$date! : DATE$

$\exists ligand? : ligands?;\ receptor? : receptors?;\ result! : results! \bullet Docking\_AutoDockVina$

ViewMolecularDockingResults_Raccoon2

$\Xi MolecularDockingEnvironment\_Raccoon2$

$results! \neq \varnothing$

MolecularDockingResultsRepository_MongoDB

$repository : (LIGAND \times RECEPTOR \times CONFIG \times DATE) \leftrightarrow RESULT$

$decisionRepository : \{PREVIOUS\_RESULT\} \leftrightarrow DECISION$

$repository \neq \varnothing$

InsertUpdateMolecularDockingResultsRepository1

$\Delta MolecularDockingResultsRepository\_MongoDB$
$l? : LIGAND$
$r? : RECEPTOR$
$c? : CONFIG$
$res? : RESULT$
$d? : DATE$

$repository' = repository \oplus \{(l?, r?, c?, d?) \mapsto res?\}$

InsertUpdateMolecularDockingResultsRepositoryMany

$\Delta MolecularDockingResultsRepository\_MongoDB$
$dockingResults? : (LIGAND \times RECEPTOR \times CONFIG \times DATE) \leftrightarrow RESULT$
$l : LIGAND$
$r : RECEPTOR$
$c : CONFIG$
$d : DATE$

$\{(l, r, c, d)\} = \mathrm{dom}(dockingResults?)$
$\forall res : dockingResults? \mid \{(l, r, c, d)\} \bullet repository' = repository \oplus \{(l, r, c, d) \mapsto res\}$

InsertUpdateDecisionRepository

$\Delta MolecularDockingResultsRepository\_MongoDB$
$previousDockingResults? : \{PREVIOUS\_RESULT\}$
$decision? : DECISION$

$decisionRepository' = decisionRepository \oplus \{previousDockingResults? \mapsto decision?\}$

SelectMolecularDockingResults

$\Xi MolecularDockingResultsRepository\_MongoDB$
$whereL? : LIGAND$
$whereR? : RECEPTOR$
$whereC? : CONFIG$
$whereD? : DATE$
$whereRes? : RESULT$
$selectResults! : (LIGAND \times RECEPTOR \times CONFIG \times DATE) \leftrightarrow RESULT$
$lig : LIGAND$
$rec : RECEPTOR$
$con : CONFIG$
$dat : DATE$

$selectResults! = \{(whereL?, whereR?, whereC?, whereD?)\} \lhd repository \lor$
$selectResults! = \{(whereL?, whereR?, whereC?, dat)\} \lhd repository \lor$
$selectResults! = \{(whereL?, whereR?, con, whereD?)\} \lhd repository \lor$
$selectResults! = \{(whereL?, whereR?, con, dat)\} \lhd repository \lor$
$selectResults! = \{(whereL?, rec, whereC?, whereD?)\} \lhd repository \lor$
$selectResults! = \{(whereL?, rec, whereC?, dat)\} \lhd repository \lor$
$selectResults! = \{(whereL?, rec, con, whereD?)\} \lhd repository \lor$
$selectResults! = \{(whereL?, rec, con, dat)\} \lhd repository \lor$
$selectResults! = \{(lig, whereR?, whereC?, whereD?)\} \lhd repository \lor$
$selectResults! = \{(lig, whereR?, whereC?, dat)\} \lhd repository \lor$
$selectResults! = \{(lig, whereR?, con, whereD?)\} \lhd repository \lor$
$selectResults! = \{(lig, whereR?, con, dat)\} \lhd repository \lor$
$selectResults! = \{(lig, rec, whereC?, whereD?)\} \lhd repository \lor$
$selectResults! = \{(lig, rec, whereC?, dat)\} \lhd repository \lor$
$selectResults! = \{(lig, rec, con, whereD?)\} \lhd repository \lor$
$selectResults! = \{(lig, rec, con, dat)\} \lhd repository \lor$
$selectResults! = repository \rhd \{whereRes?\}$

$DeepAlignCore : (RECEPTOR \times RECEPTOR) \leftrightarrow DEEP\_ALIGN\_RESULT$

$\exists\, current\_receptor : RECEPTOR;\ previous\_receptor : RECEPTOR \bullet$
$\quad \exists\, dar : DEEP\_ALIGN\_RESULT \bullet$
$\quad DeepAlignCore(current\_receptor, previous\_receptor) = dar$

DeepAlign

$SelectMolecularDockingResults$

$previous\_receptor? : RECEPTOR$

$current\_receptor? : RECEPTOR$

$DeepAlignResult! : DEEP\_ALIGN\_RESULT$

$whereR? = previous\_receptor?$

$DeepAlignResult! = DeepAlignCore(current\_receptor?, previous\_receptor?)$

$goodDeepAlignResult : (DEEP\_ALIGN\_RESULT \times USER\_INPUT) \leftrightarrow YES\_NO$

$\forall\, dar : DEEP\_ALIGN\_RESULT;\ ui : USER\_INPUT \bullet \exists\, threshold : \mathbb{Z};\ DeepScore : \mathbb{Z} \bullet$

$DeepScore \geq threshold \land goodDeepAlignResult(dar, ui) = yes \lor$

$DeepScore < threshold \land goodDeepAlignResult(dar, ui) = no$

AssessDeepAlign

$DeepAlignResult? : DEEP\_ALIGN\_RESULT$

$userInput? : USER\_INPUT$

$r : RECEPTOR$

$assessed\_receptors! : RECEPTORS$

$r \in assessed\_receptors! \land goodDeepAlignResult(DeepAlignResult?, userInput?) = yes \lor$

$r \notin assessed\_receptors! \land goodDeepAlignResult(DeepAlignResult?, userInput?) = no$

$goodDocking : (RESULT \times USER\_INPUT) \leftrightarrow YES\_NO$

$\forall\, r : RESULT;\ ui : USER\_INPUT \bullet \exists\, threshold : \mathbb{Z};\ docking\_score : \mathbb{Z} \bullet$

$docking\_score \leq threshold \land goodDocking(r, ui) = yes \lor$

$docking\_score > threshold \land goodDocking(r, ui) = no$

AssessPreviousDocking

$SelectMolecularDockingResults$

$previous\_results? : PREVIOUS\_RESULTS$

$userInput? : USER\_INPUT$

$assessedPreviousResults! : PREVIOUS\_RESULTS$

$\exists\, previous\_result : previous\_results?;\ result : RESULT \bullet$

$\{result\} = \mathrm{ran}(previous\_result) \land$

$(previous\_result \in assessedPreviousResults! \land goodDocking(result, userInput?) = yes \lor$

$previous\_result \notin assessedPreviousResults! \land goodDocking(result, userInput?) = no)$

$makeADecisionAdditionalTool : (RECEPTORS \times PREVIOUS\_RESULTS) \leftrightarrow DECISION$

$\exists\, previous\_results : PREVIOUS\_RESULTS;\ receptors : RECEPTORS;$

$lig : LIGAND;\ d : DECISION;\ con : CONFIG;\ dat : DATE \bullet$

$\forall\, previous\_result : previous\_results;\ receptor : receptors \bullet$

$\{(lig, receptor, con, dat)\} = \mathrm{dom}(previous\_result)$

$\land$

$makeADecisionAdditionalTool(receptors, previous\_results) = d$

DecisionMaker_Custom

$assessed\_previous\_results? : PREVIOUS\_RESULTS$

$assessed\_receptors? : RECEPTORS$

$decision! : DECISION$

$decision! =$

$makeADecisionAdditionalTool(assessed\_receptors?, assessed\_previous\_results?)$