Slot (třída Slot)

* Slot může být buď prázdný, nekonzistentní, nepotvrzený, nebo potvrzený
* Slot může mít žádnou, jednu, nebo více hodnot
* Potvrzený slot má právě jednu hodnotu
* Nekonzistentní slot má více než jednu hodnotu
* Hodnoty slotu jsou reprezentovány třídou Value
* Hodnota s nejvyšší confidence je první
* Hodnota s nejnižší confidence je poslední
* Přechod mezi stavy řešen podle state\_representation\_inside diagramu
* Kvůli jednoduchosti by měl být jeden obecný typ slotu
* Použití \_\_getattr\_\_, \_\_setattr\_\_, \_\_delattr\_\_ pro jednoduchý přístup k hodnotám a stavům slotu
* Přístup k libovolné hodnotě stavu
* Kontrola přijatelnosti hodnot
* Při přiřazení nové hodnoty slotu se slot se starý hodnotami odsune do History a v rámci State jej nahradí slot s novými hodnotami

Hodnota slotu (třída Value)

* Hodnota slotu má svoji confidence
* Výchozí hodnota confidence je 1.0
* Hodnota slotu je immutable

Kompozitni stav (třída State)

* Obaluje všechny sloty
* Ke Slotu se přistupuje pomocí atributu instance State
* Immutable object
* Umožňuje vrátit globální stav stavové reprezentace
* Množinu definovaných Slotů lze dynamicky měnit
* Při odstranění Slotu se Slot odsune do History, v nové instanci State již nebude jako atribut
* Instance třídy State odkazují na svojí Historii

Historie (třída History)

* Immutable object
* Prvky Historie jsou instance třídy State
* Historie musí být součástí stavové reprezentace z důvodu možnosti vrátit se do konzistentního stavu, pokud se dialog “ztratí”
* Historie se budu aktualizovat při jakékoliv změně kompozitního slotu
* Musí obsahovat celou historii dialogu

Dotazy a ukoly

* test driven programming - sepsat do jupyter notebooku pro kolaboraci
* python properties - could work
* push
* named tuple (oddedit pro Value) = data class - could work