IncQuery kérdések

* Eclipse Luna-ával használom az IncQuery-t. Okozhat-e ez bármiféle gondot? (nem találtam erre vonatkozó kompatibilitási korlátot)
* Úgy generálok Java kódot a query-kből, hogy a megfelelő projektre jobb klikkel kattintva kiválasztom, hogy "Run As" és "OSGi Framework". Ez a helyes eljárás? Mert ennek kapcsán a következő problémáim, kérdéseim merültek fel:
	+ Nem találtam a dokumentációban sehol olyan lépésről lépésre vezető leírást, mely megadta volna, hogyan kell kódot generálni (ezért a fenti kérdés).
	+ Miért próbál a generálás a workspace összes projektjéből egyszerre kódot generálni, és nem csak a kiválasztottból? Lehet-e ezt állítani?
	+ Mi lehet az oka, hogy valamennyi kódgeneráláskor kivételek sorát kapom a konzolra, de maga a kódgenerálás sikeresnek látszik (ha minden feltétel adott, hogy sikeres legyen, például nincs hiba query-kben, stb.)?
	+ Miért lehet, hogy ha letörlöm (manuálisan) a src-gen mappa tartalmát, akkor
		- hibát jelez a MANIFEST.MF fájlban, hogy hiányoznak bizonyos package-ek, még akkor, is újra (manuálisan) létrehozom üresen a package-eket, vagy törlöm a rájuk vonatkozó bejegyzést a MANIFEST.MF-ből? A hiba tehát "beragad", amin valamiért egyedül az látszott segíteni, hogy manuálisan törlöm a hibaüzenetet az Error log-ból (frissíteni vagy bezárni és újra megnyitni a projektet nem elég).
		- képtelen vagyok bármilyen módon újra kódot generálni? Azaz ha az adott projektre újból megpróbálom lefuttatni a kódgenerálást, az minden esetben eredménytelenül zárul (kivételeket sorát dobja, és nem készülnek el a fájlok).
* Mennyiben szükséges az IncQuery használatához az Xtext plugin? A dokumentációban azt olvastam, hogy igen, viszont nálam működni látszott azelőtt is, hogy feltelepítettem (kódot generálni, azt hiszem, már csak a feltelepítése után próbáltam, viszont addig semmilyen függőségi problémáról nem érkezett üzenet).
* A következő egyszerű példát írtam:

package compiler

import http://www.eclipse.org/uml2/4.0.0/UML

pattern IsClass(cl : Class) {
 Class(cl);
}

pattern StateMachine(StMach : StateMachine) {
 StateMachine(StMach);
}

* + Ehhez készítettem egy szintén leegyszerűsített Papyrus (Eclipse UML2) modellt, mely egyetlen osztályt és annak egyetlen állapotgépét tartalmazza (és egy egyszerű activity diagramot). Ha ezt a modellt szeretném a fenti lekérdezéssel vizsgálni, akkor
	+ a Query Explorer-ben az első lekérdezésre 323 találat ad, közülük olyanok, mint: DurationConstraint, EObject, EPackage, Process, Slot, UseCase, E, OpaqueAction, stb. Köztük ott van a modell egyetlen említett osztálya is, de miért jelenik a többi? (hasonló problémám merült fel egy ugyanilyen lekérdezésnél EMF-nél az EClass esetén is).
	+ Ha a generált Matcher és Match osztályok segítségével kódból végzem el a lekérdezést, a probléma hasonló, bár jóval enyhébb: két találatot ad az első lekérdezésre, nevezetesen a kérdéses egyetlen osztályt és annak állapotgépét. Tehát úgy sikerült a kívánt eredményt elérnem, hogy módosítottam a fenti első lekérdezést:

pattern IsClass(cl : Class) {
 Class(cl);
 neg find StateMachine(cl);
}

* + A StateMachine() lekérdezés mindkét esetben helyes eredményt ad.
	+ Mi lehet a felesleges találatok kiadásának oka?
* A minták között nem találtam olyat, mely részletesen bemutatná az IncQuery inkrementalitását, annak használatát. Ez csak az én figyelmetlenségemből származik, vagy tényleg nincs?
* Hatékonysági kérdések:
	+ Mennyire hatékony az IncQuery? Tehát milyen javulást érhetünk el ideális esetben egy nem inkrementális megoldáshoz képest? Ez a javulás milyen főbb paraméterek függvénye?
	+ Például modellvalidálási lekérdezések esetén a megfigyelt lekérdezések eredménye nyilvánvalóan kis elemszámú. A mi esetünkben viszont legjobb esetben is minden osztályt le kell kérdeznünk, de valamilyen formában az összes modellelem változását érzékelnünk kell, és emiatt a lekérdezések eredményének elemszáma, valamint a változások száma akár nagy is lehet. Milyen problémákat, hatékonysági gátakat jelenthet ez?
	+ Ehhez kapcsolódóan: olyan információnk van, hogy az eszköz használatának egy limite, hogy a lekérdezések eredményének elemszáma legyen az összes modellelem számához viszonyítva kicsi. Ez csak a megfigyelt lekérdezésekre vonatkozik? Tehát minden lekérdezés cache-elődik-e, vagy csak azok, melyek változásaira kíváncsiak vagyunk?
	+ Ez a fenti limit (a megfigyelt lekérdezések elemszáma legyen kicsi) globális vagy lokális? Azaz elég, ha sok lekérdezésünk mindegyike aránylag kis elemszámú, vagy a lekérdezések elemszámainak összessége legyen kicsi?
	+ Van-e rá mód, hogy például ha valamilyen formában minden activity diagram minden node-jának változásáról információt szeretnénk kapni, ezt valahogy több kisebb lekérdezésbe szedjük a korlátok megtartása érdekében (például osztályonként figyeljük őket, és ne egyetlen lekérdezés eredménye legyen az összes node)? Jelent(ene)-e ez bármilyen javulást a hatékonyságban (összefüggésben az előző kérdéssel)?
* Milyen stádiumban áll a projekt? Éles használatra késznek tekinthető, vagy bizonyos alapvető funkciók, optimalizálási kérdések még váratnak magukra?