GENERATOR	VESSEL	GHOST
	• I have a function which takes [arg types...]. • I check whether I know its argument types (reflections). If not, I query [database/parser] about the components of the Android types (what I need in the parceling (map 1) and constructor (map 2 – optional)) and create shadows containing them. • I generate recursively shadows and some Java classes. I give them to the Generator. If there is a shadow collection, the Generator needs to know how to populate the shadows.	
• I generate (X times) objects of all Java classes and return them to the Vessel.		
	• Upon acquiring the generated objects, I send (X times) a query to the Ghost. [do [method] [arg types...] [args...]]	
		• I (X times) receive the queries. For each query: ◦ I create Android objects from the shadows. At least I try. I return nulls on fail. ◦ I execute the requested method using the received and created classes. ◦ I return the results to the Vessel.

A general idea about the inter-application communication. Probably will be updated.