In our proposal interface components are not cooperating, but they are competing to gain interface resources. This competition is ruled by a marketplace in which interface assets are regularly on sale, and applications attempt to acquire them spending credits.

Coordination is not coming out by cooperation. Instead, it is expressed as equilibrium among competitors. The problem of agents competing for limited resources is quite old, and widely studied by Economics to model social behavior. Stock Exchange provides a good example from which to take a move. In this market, trades are based on an auction market paradigm where a potential buyer bids a specific price for a stock and a potential seller asks a specific price for the stock. When the bid and ask prices match, a sale takes place on a first-come first-served basis if there are multiple bidders or askers at a given price. Exchange is aimed at facilitating the trading of securities between buyers and sellers.

This inspires a novel approach in conceiving user interfaces. Not a set of resources used by applications according only to their needs, but a set of assets that must be acquired in order to perform application functionalities. The UI real estate is made of different assets, such as screen regions, audio channels, keyboard, pointer and other input devices.

The marketplace (Fig. 1) is where the demand and the offer for UI assets meet. The UI orchestration is reached according to rules governing the marketplace. In First-Price Sealed-Bid (FPSB) auction, participants make a simultaneous bid without knowing the bids made by the others.

Applications gain the control of UI assets for a limited time. When the asset is released it goes on the market for sale. Applications decide which assets to make bid on. The Marketplace decides the auction winner. Applications holding assets also pay a maintaining tax. Taxes and auction revenues are given to the Regulator, that is in charge of redistributing collected credits according to two policies: Capitalism, to reward most used applications, and Welfare State, to assist applications with lower credit availability.

The scenario we considered has an initial activity for choosing a video and start to watch it, with mail messages arriving in the mailbox. Later chat session starts, whilst the movie ends.

The approach we are presenting, provides benefits from both the application user and designer point of view. Designers are not forced to think about how to organize components in their application, and applications are not forced to cooperate within a framework in order to get the components orchestration. Designers are only demanded to provide a bidding strategy to their applications, and asset assessment criteria on which to base the bidding decision. Indeed, some assets could be more valuable than others according to the application needs.

On the other side, users can interact with a set of applications without being obliged to control and orchestrate explicitly the components deployed at the interface. Applications becomes more independent on GUI layout. They can automatically adapt to different layouts, as they are able to estimate how assets are arranged together. We believe that the Efficient Market Hypothesis [4] can be applied to the UI marketplace. Finding design rules able to make the market efficient is important, as it makes the application behavior independent from each other, and it allows to build robust bidding strategies.

The approach can be applied to other problems, where a demand of resources must face limitation of them, such as the service orchestration and the job allocation in Grid Computing.

**REFERENCES**

5. A. Ganek and T. Corbi. The dawning of the autonomous...