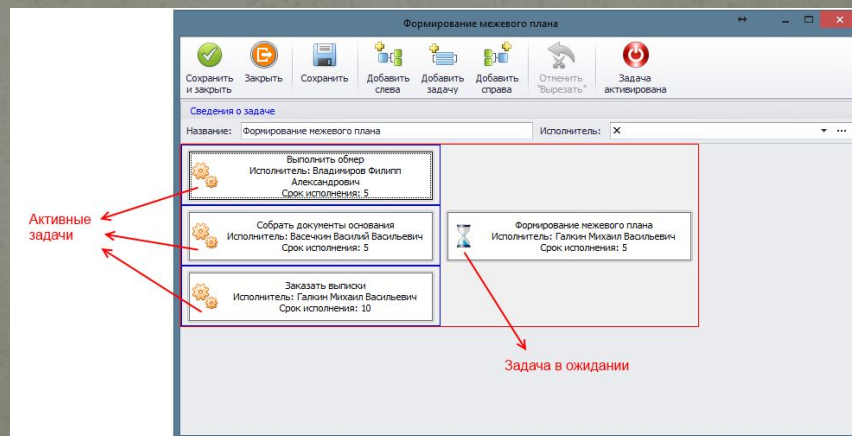


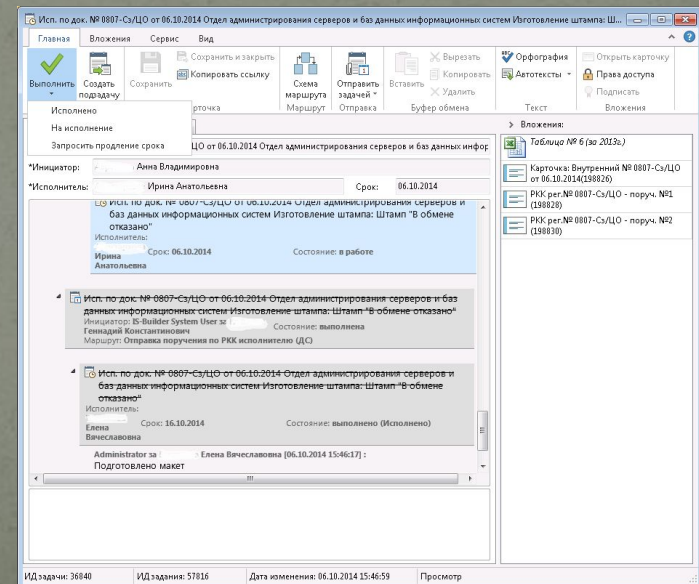
ОПЕРАТОРЫ СВЕДЕНИЯ ЗАДАЧИ К ПОДЗАДАЧАМ



Оператор сведения задачи к подзадачам преобразует описание задачи во множество результирующих, или дочерних, описаний задач. Это преобразование таково, что решение всех дочерних задач обеспечивает решение исходной родительской задачи. Когда множество дочерних задач состоит из одного элемента, мы имеем простейший случай замены одной задачи другой, ей эквивалентной.



Для данного описания задачи может существовать много операторов сведения, каждый из которых применим. Применение каждого такого оператора порождает альтернативные множества подзадач. Некоторые из этих подзадач могут оказаться неразрешимыми, так что нам придется перепробовать несколько операторов, чтобы построить такое множество, все члены которого разрешимы. Таким образом, снова возникает задача перебора.



Один класс задач связан с доказательством того, что определенные утверждения истинны. Пусть — утверждение, истинность которого мы хотим доказать, множество посылок, которые предполагаются верными. Тогда под (читается « S при данном T ») мы будем понимать задачу доказательства утверждения исходя из посылок T .

Общая схема для сведения к подзадачам задач такого вида, состоит в том, чтобы ввести в исходную задачу новые посылки, а затем сформулировать дополнительные задачи на доказательство этих новых посылок. Так, при редукции задачи мы добавляем, скажем, посылок и получаем следующее множество результирующих задач:

$$\begin{array}{l} S | T, X_1, X_2, \dots, X_N \\ X_1 | T, X_2, \dots, X_N \\ \vdots \\ X_N | T \end{array}$$

Где — это добавочных посылок. Часто этот оператор редукции задачи применяется так, чтобы в каждый момент времени добавлялась лишь одна посылка. Тогда сводится к множеству

Символы посылок можно было бы рассматривать как переменные, принимающие значения из некоторого множества посылок.

Тогда каждое возможное значение этих переменных соответствовало бы применению отдельного оператора сведения задачи. Этим переменным могут быть сразу же приданы определенные значения в виде конкретных посылок (содержащих, возможно, новые переменные), но вместо этого их можно оставить в виде переменных, имея в виду, что конкретные значения могут быть им приданы в процессе дальнейшего сведения. Позже мы рассмотрим некоторые из предлагаемых подходов к выбору конкретных значений для посылок. Часто нам будет нужно в качестве частных значений выбрать более чем одно множество посылок, так что наше доказательство может вернуться назад и пойти по другим альтернативным направлениям.