

Accessibility: взгляд изнутри

Михаил Пожидаев

Linux Format RE, 2013, №1

В наше время представить себе полноценную жизнь без компьютера практически невозможно. И, естественно, Accessibility, то есть облегчение доступа к компьютеру пользователям, имеющим физические ограничения, является не последней заботой также и операционной системы *GNU/Linux*, о чем рассказывалось в LXF146. В частности, приводилось описание функций утилиты чтения экрана *Orca*, предназначенной для людей с проблемами зрения. Ниже мы предлагаем читателю более детально познакомиться с компонентами, лежащими в основе вспомогательных технологий *GNU/Linux*, а также проанализировать возможное развитие этого направления в будущем.

Экранный чтец *Orca* разработан для представления оконного интерфейса *Gnome* в речевом виде. При каждом действии пользователя он озвучивает элемент, на котором остановился фокус ввода. Не глядя использовать указатель мыши обычно достаточно тяжело, поэтому все команды передаются при помощи клавиатуры. Основная работа по организации доступной среды выполняется двумя крупными сервисами, взаимодействие которых *Orca* поддерживает с учётом параметров, установленных пользователем.

Один из этих сервисов — так называемый

речевой сервер. Он занимается обслуживанием речевых синтезаторов, выполняет их запуск и организует воспроизведение звукового потока. Термин “сервер” правильно отражает назначение приложения, поскольку возможно существование нескольких источников речи (клиентов), для которых необходимо правильно организовать совместную работу. Тем не менее, речевой сервер не является сервером в полном смысле этого слова, потому что клиенты не изолированы друг от друга, допускается воспроизведение только одного фрагмента текста в каждый момент времени, и клиенты вправе отменять выполнение заданий друг друга. Кроме *Orca*, источниками речи могут быть различные системные оповещения, включающие уведомления о подключении съемных носителей информации, речевые часы и пр.

Другой сервис, называемый *AT-SPI*, играет более важную роль в процессе предоставления вспомогательных технологий. Название расшифровывается как Assistive Technology Service Provider Interface (Интерфейс поставщика сервиса вспомогательной технологии). *AT-SPI* занимается сбором полной информации о пользовательском интерфейсе приложений, включая сведения об открытых окнах, расположении элементов управления и других деталях, позволяющих получить представление об изображении на экране. Реализация *AT-SPI* подразумевает решение двух крупных технических задач: внедрение в библиотеки построения оконного интерфейса компонента для получения необходимой информации и сбор полученной информации в рамках одного про-

цесса для последующей трансляции в речевую форму. Для решения первой задачи требуется подготовить дополнения ко всем популярным оконным библиотекам, а также ко всем приложениям, использующим собственные реализации оконных виджетов. Поскольку развитие *AT-SPI* велось преимущественно в рамках *Gnome Accessibility Project*, то первой библиотекой, получившей необходимую поддержку, стала *GTK+*. Со временем подобные дополнения появились в браузере *Firefox*, офисном пакете *OpenOffice.org* (а впоследствии и в *LibreOffice*), в *Java/Swing* и, наконец, в *Qt4*. Таким образом, вспомогательные технологии будут доступны в большинстве приложений, получивших распространение в *GNU/Linux*.

Для централизованной обработки полученной информации используется специальное приложение *at-spi-registryd* (см. рис. 1), запускаемое при старте оконной системы и функционирующее в фоновом режиме. Основная сложность в реализации концепции *AT-SPI* заключается в том, что *at-spi-registryd* и компоненты, интегрированные в оконные библиотеки, исполняются в разных процессах. Обмен информацией напрямую невозможен, и применяются средства организации межпроцессного взаимодействия. Первая версия *AT-SPI* использовала в своей работе *Corba/Bonobo*, но в дальнейшем от этого выбора пришлось отказаться, поскольку в *Gnome 3* использование *Corba/Bonobo* было признано устаревшим, и на его замену пришел сервис *D-Bus*. Обычно при использовании *D-Bus* задействуется одна из двух так называемых шин: либо системная шина,

функционирующая для обмена информацией среди всех приложений ОС, либо сессионная шина, при помощи которой выполняется взаимодействие приложений внутри одной сессии пользователя. Для работы *AT-SPI* был выбран третий подход, заключающийся в запуске отдельной копии сервера *D-Bus*, предназначенной для обслуживания взаимодействия приложений только в рамках задач *AT-SPI*. Такой выбор призван повысить производительность системы, поскольку проблемы скорости работы *AT-SPI* очень актуальны: информации об интерфейсе приложений очень много, и передача через *D-Bus* занимает значительное время. В завершение описания концепции *AT-SPI* заметим, что утилита *ChromeVox*, предлагаемая компанией *Google* для работы незрячих и слабовидящих людей в браузере *Chrome*, не использует функций *AT-SPI*, хотя это не исключает появление поддержки в будущем.

Концепция *AT-SPI*, обещающая достаточно широкие возможности для работы незрячих и слабовидящих людей с оконными приложениями, продумана очень тщательно, но существуют и альтернативные подходы. Например, интересный вариант основан на добавлении вспомогательных функций в текстовый редактор *GNU Emacs*. Как известно, круг задач, решаемых при помощи *GNU Emacs*, не ограничивается текстовым редактором. За счет многочисленных дополнений пользователи имеют возможность работать с электронной почтой, с удобным файловым менеджером, составлять записи в дневнике, просматривать календарь, вести разработку ПО и пр. Доступен упрощенный вариант web-

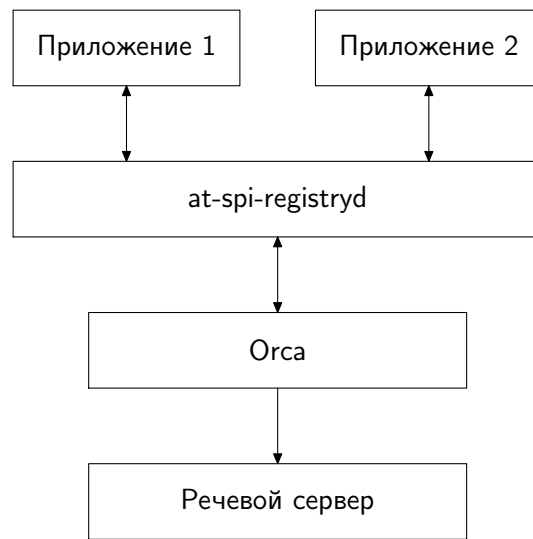


Рис. 1: Структура AT-SPI

браузера, основным ограничением которого является невозможность исполнения *Java Script*. Дополнения к *GNU Emacs* подготовлены таким образом, что все рабочие объекты представлены в текстовом виде. Именно это сыграло решающую роль в популярности подхода, основанного на *GNU Emacs*.

Необходимый набор вспомогательных функций для *GNU Emacs* поставляется в пакете *emacspeak*. Основная задача *emacspeak* — выполнять речевые оповещения в ответ на все действия пользователя. При перемещении курсора вверх-вниз *emacspeak* произносит полный текст строки, в случае движения вправо-влево — символ в новом положении курсора. Рабочее пространство может разбиваться на несколько окон, в каждом из которых отображается некоторый документ. Помимо приложений из состава самой среды *GNU Emacs*, пользователь может использовать утилиты командной строки, запуск которых выполняется при помощи терминала, оснащенного речевым интерфейсом. Речевой вывод, формируемый *emacspeak*, перенаправляется в речевой сервер, функции которого сов-

падают с функциями речевого сервера, используемого для работы утилиты *Orca*, упомянутой выше.

Основное преимущество работы в *GNU Emacs* заключается в высокой скорости взаимодействия с ПК незрячего или слабовидящего человека, которая достигается за счет отсутствия необходимости навигации по элементам оконного интерфейса, ориентированного изначально на представление информации в графическом виде и управление при помощи мыши. Важную роль в достижении высокой скорости играют и широкие возможности назначения "горячих» клавиш". Гармоничным дополнением к среде *GNU Emacs* в случае использования незрячим или слабовидящим человеком является доступность таких пакетов, как *LaTeX* (со входящими в него *MetaPost* и *Beamer*), *Lilypond* и *GNUPlot*. Все эти пакеты объединяет возможность подготовки материалов в текстовом виде с последующим экспортом в PDF-файл. Обладая необходимыми знаниями и навыками, пользователь может работать над большими текстами (книгами, диссертациями), нотными

партитурами и презентациями. Работа производится незрячим человеком в основном самостоятельно, но последующая визуальная проверка остается необходимой.

В сообществе *GNU/Linux* накоплено много опыта работы при помощи речевого интерфейса — как с использованием оконной среды, так и в *GNU Emacs*. *AT-SPI* имеет качественную реализацию и может стать стандартом вспомогательных технологий для всех оконных окружений, а не только для *Gnome*. Поддержка *AT-SPI* обсуждается в сообществах *Xfce*, *KDE* и даже *Tizen*, но в каждом случае требуется подготовка новой утилиты на замену *Orca*, которая имеет жесткие зависимости с библиотеками *Gnome*. Главная проблема этого направления в том, что разработчики приложений нередко прибегают к использованию собственных элементов управления, забывая о необходимости совместимости со вспомогательными технологиями. При работе с *GNU Emacs* тоже существуют трудности. Основная проблема заклю-

чается в том, что *GNU Emacs* изначально ориентирован на опытного пользователя, уровень знаний которого приближается к уровню разработчика ПО. Пользовательские настройки сохраняются в виде скриптов на языке *Lisp*, а многие задачи требуют подготовки сценариев *Bash*. Ситуацию осложняет невозможность перевода речевого вывода на национальные языки, это делает массовое распространение очень затруднительным, но приведенные недостатки не вытекают из самой концепции представления рабочих объектов в текстовом виде, которая позволила выработать подход к работе незрячего или слабовидящего человека на ПК со скоростью, сравнимой со средней скоростью работы людей в различных коллективах. Дальнейшее развитие этого направления дает надежду на появление новых продуктов, лучше приспособленных для использования в рамках учебного или производственного процесса, чем подходы с озвучиванием оконного интерфейса. Присоединяйтесь! |