**로봇 소프트웨어 배포 사례**

소속 : (주) 클로봇 미들웨어개발팀

현업에서 다양한 로봇 프로젝트들을 수행하면서 개발자로서 겪었던 소프트웨어 업데이트 경험담을 공유합니다. 로봇 통합 시스템을 구축하는데 발생했던 의존성 문제와 최신 버전의 소프트웨어를 배포하고자 할 때 겪었던 문제점을 이야기하고, 이로서 Ubuntu Snap 을 활용하면서 얻을 수 있는 장점에 대해 발표합니다.

<발표 내용>

  안내, 이송, 돌봄, 순찰 분야에 이어 물류, 건설 현장 자동화 등을 목표로 로봇의 다양한 적용 사례가 꾸준히 증가하고 있습니다. 우리는 현업에서 여러 로봇 프로젝트(집안에서 사용자에게 홈 라이프와 케어 기능을 제공하는 프로젝트, 오피스, 관공서 등 여러 층을 이동하며 물품 이송 서비스를 제공하는 멀티 로봇 프로젝트, 실외 환경을 이동하며 순찰 기능을 제공하는 프로젝트 등등)들을 수행하고 있으며, 이렇게 다양한 로봇들의 동작을 안정화하고 고도화하기 위해 소프트웨어를 업데이트해야 하는 상황이 빈번히 발생합니다. 발표에서는 로봇 소프트웨어를 배포하기 위해 그동안 해왔던 방식과 앞으로의 방향을 공유합니다.

1. 설치자가 직접 로봇 소프트웨어를 업데이트 하는 방식.

소스코드를 로봇에서 직접 빌드했던, 가장 단순한 방식입니다. 이 경우, 휴먼 에러가 높아 설치자의 실수 등으로 시스템에 문제가 발생하기도 하였습니다. 또한, 현장에 직접 방문하여 로봇에 작업해야 하는 등 인력, 시간, 비용의 측면에서 많은 리소스가 소모 되었습니다.

1. Debian 패키지로 소스코드를 빌드하여, 이를 설치하는 방식.

소스코드를 직접 빌드하지 않고, Debian 패키지로 만들거나, ROS 의 경우 ROS BuildFarm을 이용하여 apt로 설치할 수 있게 하였습니다. 이 경우, 소스코드를 빌드했던 것처럼 휴먼에러는 줄일 수 있었지만, 다른 패키지들간의 의존성 문제는 해결하기 어려웠습니다.

1. 로봇 기동 시 OTA(Over-the-air) 서버에 연동하여, 버전 정보를 확인하고 업데이트를 수행하는 방식.

OTA 서버를 구성하는 경우, 적합한 배포 정책을 통해 효율적이고, 안정된 업데이트를 제공하는 경우도 있었지만(물류 분야 등 로봇의 수가 많은 경우), 소규모의 프로젝트를 다수 수행하는 경우, OTA서버를 따로 구성하기 어려운 상황이 있습니다. 또한, 운영체제 등 시스템 사양이 변경되어야 하는 경우, 기존 소프트웨어를 그대로 사용하지 못하고, 새로 개발해서 적용해야 하는 문제는 여전히 있었습니다.

1. Docker 등 컨테이너로 구성하여 소프트웨어를 업데이트 하는 방식.

현재 가장 많이 활용하고 있는 방식입니다. 컨테이너로 구분하여 로봇 통합 시스템 구성 시, 의존성을 매우 낮출 수 있었으며, 소프트웨어 업데이트에 소요되는 리소스를 줄일 수 있었습니다. 하지만 컨테이너의 용량이 커지는 문제로 더욱 경량화 된 배포 방식이 무엇인지에 대해 고민을 하게 되었습니다.

그래서, 우리는 Ubuntu Snap 을 활용하여 로봇 기동 시, 자동 업데이트를 수행하도록 방향을 수립하고 있습니다. Snap은 배포에 초점을 두어 GUI를 통한 쉬운 설치 방법 제공과 자동 업데이트로 사용자가 로봇 소프트웨어를 최신으로 유지하기 쉬울 뿐만 아니라 버전 롤백 또한 지원하고 있습니다.  이의 적용 사례에 대해 소개하고자 합니다.

* 전달하고 싶은 ‘대상’
* 로봇 소프트웨어 개발 후, 향후 업데이트 배포 계획을 고민하고 있는 로봇 회사 직원

:  로봇 시스템은 다양한 목적에 맞게 로봇의 소프트웨어 구성 또한 다릅니다. 각 구성 요소(구동부, 센서, 주행 제어, 로봇 팔 제어, 통신, 인공지능, 서비스 등등) 마다 발생하게 되는 배포 문제를 고민하고 있는 분들에게 ubuntu snap과 같은 모듈화된 배포방식의 장점을 설명합니다.

* 소규모 프로젝트에서 로봇 소프트웨어를 구성하고 배포하려는 개발자

: 소규모 프로젝트에서 로봇 소프트웨어를 지속적으로 배포하기 위한 기반을 만드는 것에 큰 부담을 느낄 것입니다. 다양한 프로젝트에서 배포 경험을 공유하여 효율적인 운영 프로세스를 바라는 분들에게 도움이 될 것입니다.

* 발표를 통해서 청중이 어떤 것을 얻을 수 있을까요?

현업에서 로봇 운영 시에도 예상하지 못했던 이슈를 해결하기 위해 소프트웨어를 업데이트 하는 상황, 고객의 요구에 의해 추가적인 기능을 개발하여 소프트웨어를 업데이트 하는 상황 등 지속적인 배포를 경험하게 됩니다. 이의 배포 경험담을 공유하면서 비슷한 고민을 하고 있거나, 자신의 시스템에 배포 프로세스를 적용해보고자 하는 분들에게 도움을 줄 수 있습니다.