

GNU Parallel 조사 - 찬혁

공식 문서 요약

1. GNU Parallel의 개요

- 정의: GNU Parallel은 동시에 여러 명령을 실행하여 데이터를 병렬로 처리할 수 있게 해주는 셸 도구입니다.
 - 이는 CPU의 멀티코어 기능을 활용하여 처리 성능을 향상시킬 수 있습니다.
- 주요 특징:
 - 간단한 문법으로 합쳐서 명령을 실행할 수 있음
 - 다양한 출력 형식 지원
 - SSH를 통한 원격 처리 가능

2. 설치 방법

- 소스에서 설치: 문서에는 소스 코드로부터 GNU Parallel을 설치하는 방법이 소개되어 있으며, GNU 웹사이트에서 최신 버전을 다운로드할 수 있습니다.
- 패키지 관리 시스템: 여러 리눅스 배포판에서 제공하는 패키지 관리자(yum, apt 등)를 사용하여 손쉽게 설치할 수 있습니다.

3. 기본 사용법

- 명령어 형식: 기본적인 사용 형식 및 여러 명령을 병렬로 실행하는 방법이 설명되어 있습니다.
 - 이 명령은 "A", "B", "C"를 각각 병렬로 출력합니다.

SHELL

```
parallel echo ::: A B C
```

- 옵션 및 플래그: 다양한 옵션을 통해 세부 기능을 설정할 수 있으며, 이를 통해 출력 형식이나 실행 환경을 조정할 수 있습니다.

4. 고급 기능

- 입력 및 출력 제어: 입력 파일에서 데이터를 읽어와 병렬 처리할 수 있으며, 결과를 다시 출력 파일로 저장하는 방법도 포함되어 있습니다.
- 모니터링 및 성능 최적화: 실행 중인 작업을 모니터링하고, 성능을 개선하는 팁이 제공됩니다.

5. 예제 및 사용 사례

- 실제 사용 예제: 다양한 실전 사례를 통해 어떻게 GNU Parallel을 활용할 수 있는지가 설명되어 있습니다.
 - 예를 들어, 데이터 처리, 웹 크롤링, 대규모 계산 등을 위한 활용법이 있습니다.
-

4. Input Sources

- GNU Parallel은 여러 가지 입력 소스를 사용하여 작업을 병렬로 실행할 수 있습니다.
- 여기에는 파일, 명령어의 출력, 사용자 입력 등이 포함됩니다.
- 다음은 각 입력 소스를 사용하는 방법에 대한 설명입니다.

4.1 파일에서 입력 받기

- 파일로부터 입력을 받을 때, `cat filename | parallel` 명령을 사용할 수 있습니다.
- 이때 `filename` 파일의 각 줄이 개별 작업으로 처리됩니다.

4.2 명령어의 출력 사용

- 명령어의 출력을 입력으로 사용할 때는 다음과 같이 할 수 있습니다:

```
seq 10 | parallel echo {}
```

- 이 명령은 `seq 10`의 출력을 받아 각 숫자를 `echo` 명령에 전달합니다.

4.3 인수로 직접 입력 제공

- 입력을 인수로 직접 제공할 수도 있습니다.
- 예를 들어, 다음과 같은 방식으로 여러 입력값을 지정할 수 있습니다:

```
parallel echo ::: 1 2 3 4
```

- 이 명령은 `1`, `2`, `3`, `4`를 각기 다른 `echo` 작업으로 실행합니다.

4.4 조합된 입력

- 입력을 결합하여 처리하는 것도 가능합니다.
- 예를 들어, 두 개 이상의 목록을 결합하려면 `:::`를 사용합니다:

```
parallel echo {1} {2} ::: a b ::: 1 2
```

- 이렇게 하면 `{1}` 자리에는 `a`, `b`가, `{2}` 자리에는 `1`, `2`가 각각 들어가 모든 가능한 조합을 실행하게 됩니다.

4.5 동적 입력

- GNU Parallel은 동적으로 입력을 받을 수도 있습니다.
- 특정 조건에 맞는 입력을 추가로 제공하거나 변경할 수 있어 다양한 작업 흐름에 유용하게 적용할 수 있습니다.

5. Build the command line

기본적인 개념

- GNU Parallel은 여러 개의 작업을 동시에 수행할 수 있게 해 주는 도구입니다.
- 사용자는 명령어 단위로 작업을 설정할 수 있으며, 각 작업은 독립적으로 실행됩니다.

명령어 줄 생성

- 명령어 줄은 작업의 입력과 출력 형식을 정의합니다.
- 다음은 기본적인 명령어 형식입니다:

SHELL

```
parallel [옵션] 명령어 ::: 입력리스트
```

- 여기서 ::: 구분자는 처리할 입력 항목을 나열하는 데 사용됩니다.

입력 데이터 정의

- 입력 파일 또는 표준 입력으로부터 데이터 리스트를 읽을 수 있습니다.
- 예를 들어, 텍스트 파일에서 각 줄을 입력으로 사용할 수 있습니다.

옵션 사용

- GNU Parallel의 각종 옵션을 통해 사용자 편의에 맞게 다양한 설정이 가능합니다.
- 예를 들어, -j 옵션을 사용하여 동시에 실행할 작업의 수를 설정할 수 있습니다.

예제

- 기본적인 사용 예:

SHELL

```
parallel echo ::: "Hello" "World"
```

- 이 명령어는 "Hello"와 "World"를 각각 병렬로 출력합니다.

고급 기능

- --dry-run: 명령을 실행하지 않고 무엇이 실행될 것인지 확인할 수 있는 옵션입니다.
- --jobs: 최대 실행할 작업의 수를 조정할 수 있습니다.