

## 메쉬 센서 네트워크 기술 동향 분석

### Wireless mesh sensor network, the new emerging low-rate wireless network technology

강정훈, 이민구, 임호정, 윤명현, 유준재  
전자부품연구원

Jeonghoon Kang, Mingoo Lee, Hojung Lim, Myunghyun Yoon, JunJae Yoo  
Korea Electronics Technology Institute

**요약** - 메쉬 센서 네트워크 기술은 새로운 컴퓨팅 패러다임인 유비쿼터스 컴퓨팅의 진입 기술로 중요한 의미를 갖는다. 이런 새로운 컴퓨팅 패러다임의 기술이 성공적으로 발전하기 위해서는 기반이 되는 진입기술의 확산이 중요하다. 본문에서는 현재까지 진행된 메쉬 센서 네트워크 기술 동향과 실제로 구현된 응용 서비스들을 분석한다. 그리고 메쉬 센서 네트워크에 대한 개념 정립과 실제 구현을 동시에 진행하고 있는 TinyOS 프로젝트에서 공개된 메쉬 센서 네트워크 구조 등, 주요 기술 부분과 응용 서비스 기술을 분석하여 지금까지의 메쉬 센서 네트워크 기술개발 상황과 향후 산업화 가능성을 파악한다.

**키워드** : wireless, mesh sensor network, low power, multi-hop, ubiquitous computing

#### 1. 서론

메쉬 센서 네트워크 기술은 새로운 컴퓨팅 패러다임인 유비쿼터스 컴퓨팅의 진입 기술로 중요한 의미를 갖는다. 이런 새로운 컴퓨팅 패러다임의 기술이 성공적으로 발전하기 위해서는 기반이 되는 진입기술의 확산이 중요하다. 본문에서는 현재까지 진행된 센서 네트워크 기술 동향과 오픈소스 프로젝트로 진행 중인 TinyOS를 이용하여 구현된 응용 서비스들을 분석한다. TinyOS는 새롭게 떠오르고 있는 센서 네트워크에 대한 개념 정립과 실제 구현을 동시에 진행하고, 그 결과를 공개하여 센서 네트워크의 확산을 이끌고 있다. 실제 응용 서비스를 구현하고 테스트 하고 때문에 가장 앞서나가고 있는 메쉬 센서 네트워크 기술이라 할 수 있다. 현재까지 개발되어 공개된 메쉬 센서 네트워크의 주요 기술 부분과 응용 서비스를 분석하여 지금까지의 기술개발 상황과 향후 산업화 가능성을 파악한다.

#### 2. 연구 개발 동향

근래에 제조업자들은 센서들을 많은 가정기나 자동차나 부속품들에 부착하였다. 그러한 놀랄만한 성장은 물리적인 데이터들 위치들이 전송하고 저장하고 처리할 수 있는 정보의 단위로 변환하는 것처럼 센서가 읽은 값을 다른 장치들에 무선으로 전송하는 것이다[1][2][3].

무선 센서 네트워크의 좀 더 전형적인 운영방법은 이러한 자원들의 작은 부분을 가지고 AA 크기 배터리 두 개로 일년 이상 동작한다. 더욱 이러한 응용 서비스는 복잡한 인간의 상호작용에 집중 하지 않고 물리적인 세계의 구조적인 상호작용에 초점을 맞춘다. TinyOS 같은 오픈소스 개발자들은 이러한 응용 서비스를 고려하여 개발하

고 있다.

TinyOS는 제한된 물리적 자원을 가지고 중요한 동시 처리를 조절할 수 있는 통합적인 시스템 제어 기법에 필요한 기본 기능을 제공한다. 소프트웨어 컴포넌트들과 기초가 되는 운영체제는 구체적인 이벤트 기반의 기능을 지원한다. 최저 레벨의 컴포넌트들은 물리적 하드웨어를 추상화하고 비 규칙적인 동작의 물리적 인터럽트를 전달한다.

각 컴포넌트 들은 다른 컴포넌트들에서 정해진 이벤트 들이나 신호의 동작을 제어하지만, 그것들은 다음 행할 이벤트를 위해 대기하는 동안에는 프로세서를 사용할 수 없다. 각각의 응용 서비스들은 단지 그것이 요구하는 컴포넌트들만을 포함한다. 예를 들면 컴포넌트들의 작은 스택들은 센서가 감지한 값을 처리한다. 가장 낮은 컴포넌트들은 원래 읽은 값을 얻기 위해 ADC 컨버터를 조절하는데, 반면에 상위 수준의 컴포넌트들은 응용 서비스를 위한 데이터들을 선별적으로 추출한다.

네트워크는 데이터를 수신하는 노드들이 인식할 수 있는 패킷으로 구성되는 낮은 레벨들이 무선채널을 획득하는 작업을 처리하는 위치에서 좀 더 복잡한 스택과 연관된다. 이러한 컴포넌트들은 입력되는 패킷들의 도착을 감지하고 그것들을 입력 버퍼에서 처리할 뿐만 아니라 또한 오류 검출과 채널 스케줄링을 수행한다.

상위 레벨들은 버퍼 관리 및 인증, 응용 애플리케이션들 사이의 네트워크를 다중 송수신 작업을 처리한다. 일반적인 상위 레벨 컴포넌트는 취득된 센싱 값들의 데이터를 수신하고 처리할 것이고, 그때 네트워크에 중요한 신호를 전달할 것이다. 두 번째의 컴포넌트는 그러한 신호를 받을 것이고, 라우팅 구조를 유지하고 다음 홉 사이에서 그 데이터를 수신하여 게이트웨이로 재전송한다.

TinyOS는 태스크의 동시성을 지원한다. 단순한 큐를 기반으로 태스크 스케줄러는 사용자에게 태스크로 함수를 호출할 수 있도록 지원하며, 인터럽트 처리 루틴과 연동되어 작은 크기의 스케줄러로 무선 네트워크 기능을 동작 시키도록 지원한다[4].

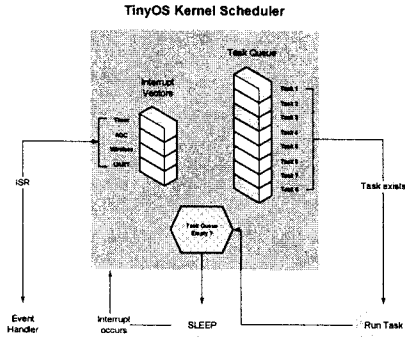


그림 1. TinyOS 태스크 스케줄러

버클리 모트와 TinyOS는 시스템들의 이슈를 연구하고 초기에 실제 테스트 응용 애플리케이션을 배치하기 위해 널리 사용되고 있다. 마이크로컨트롤러는 적당한 양의 RAM과 프로그램을 위한 저장 공간을 제공하고 내부에 ADC를 포함한다. 넓은 주파수 범위의 모델을 가지고 있는 간단하면서 주파수에 민감한 무선 송수신 모듈은 개발자가 네트워크를 구성하기 위해 사용하는 연결성을 제공한다. 칩과 분리된 플래쉬 메모리는 네트워크를 통해서 그것이 전송하는 동안의 프로그램과 칩에 붙어있는 RAM을 초과하여 임시 저장하는 데이터 모듈을 저장하기 위한 메모리를 제공한다. 다수의 센서 보드들은 이러한 플랫폼을 위해 고안되어 왔다. TinyOS를 지원하는 하드웨어 플랫폼인 MOTE는 1999년부터 다양한 형태로 연구 개발 되어왔다.

<표 1> Mote 하드웨어 플랫폼 종류

Mote	WeC	rene	dot	mica	mica2	mica2dot	iMote	bitNode
년도	1999	2000	2001	2002	2003	2003	2003	2003
프로세서 속도	4 Mhz			7 Mhz	4 Mhz	12 Mhz	7 Mhz	
플래쉬 (KB)	8	8	16	128	128	128	512	128
램(KB)	0.5	0.5	1	4	4	4	64	4
라디오 (kBaud)	10	10	10	40	40	40	460	460
라디오 종류	RFM			ChipCon	ChipCon	ZeevoBT	EricsonBT	
마이크로 컨트롤러	Atmel						ARM	Atmel
확장	불가능	가능	불가능	가능	가능	가능	가능	가능

인텔 아이모트[5]는 기능이 높은 ARM 마이크로 프로세서와 저장공간이 하나의 패키지로 결합되어진 무선 송수신기를 함께 탑재하고 있는 상용화된 칩을 사용하여 최근에 완성된 플랫폼이다. 무선 송수신기는 노트북과 휴대

폰 폰에 널리 적용되고 있는 블루투스 표준을 적용시켰다. 무선 통신은 높은 범위에서 동작하고 정교한 주파수 호핑 프로토콜을 가지고 있다. ARM 프로세서는 블루투스 라디오를 운영하고 무선 통신 모듈부터 시리얼 포트까지 패킷을 전달하는 기능을 한다. 아이모트에서 동작하는 TinyOS는 다양한 센서들과 라우팅 기능들을 제공하고 상위레벨의 정보를 처리하고 전력소비를 관리하는 하나의 통합된 시스템을 제공하면서 ARM 프로세서를 직접 구동한다. 낮은 수준의 TinyOS 컴포넌트들의 대부분은 하드웨어에 직접 연동된다. 특징은 스마트 dust 프로젝트에서 개발된 저전력의 ADC와 효율적인 라디오 모듈을 포함한다. 전체적인 디자인은 단지 5 제곱 밀리미터의 영역만을 차지한다. 그것은 1% 활성화 비율로 AA 배터리 두 개를 사용하면 100년 동안 동작할 수 있다고 평가된다.

하나의 네트워크는 다른 노드들에 연결하는 다중의 링크들을 가지고 각각 많은 노드들로 구성된다. 정보는 생성지점부터 사용되는 지점까지 하나의 라우트를 통해 홉에서 홉으로 이동한다.

인터넷 같은 서로 연결된 네트워크에서 각각의 라우터는 라우팅 그래프를 형성하면서 다른 라우터들의 하나의 구체적인 집합에 연결한다. 무선 센서 네트워크에서 각 노드들은 노드 근체에 통신링크들의 한 집합을 제공하는 무선 송수신기를 가지고 있다. 정보를 교환함으로써, 노드들은 그들의 이웃 노드를 발견하고 응용 애플리케이션의 요구에 따라 데이터를 라우팅하는 방법을 결정하는 분배 알고리즘을 수행한다. 비록 물리적인 위치가 연결성을 주로 결정하지만 장애물이나 간섭이나 환경적인 요소, 안테나 방향 그리고 이동성 같은 변수들은 연결의 우선순위를 결정하기 어렵게 만든다.

무선 센서네트워크의 네트워크 기능은 레이어를 나누어 처리하여 성능을 향상 시킬 수 있다. 가장 낮은 레이어는 물리적인 라디오 장치를 제어한다. 라디오 들은 전송개체 제어가 자연에 의해 이루어지는데, 하나의 노드가 전송할 때 다른 노드들의 집합은 신호를 수신한다. 그렇지 않으면 동시에 다른 노드가 전송하는 신호에 의해 잘못 선택되어 질 것이다. 라디오 전송 채널에서 충돌을 피하기 위해 링크 레이어는 채널을 청취하고 전송 채널에 다른 무선 전송이 없을 때만 전송한다. 그것은 라디오 신호에서 부호화된 패킷으로부터 구조적인 비트들의 배열로 전송한다.

전송하지 않을 때, 노드들은 채널을 샘플링하고 수신기가 송신자의 시간을 가지고 노드 자체를 배열하는 것을 허용하는 패킷의 시작부분에서 특정한 데이터 수신을 위해 항상 수신을 수행한다. 해당 패킷 레이어는 버퍼들을 관리하고, 라디오 장치에서 패킷들의 일정을 정하고 오류를 검출하거나 정정하고, 패킷 손실을 조절하고 시스템이나 응용 애플리케이션에 패킷들을 신속히 처리한다.

센서 네트워크 사용자들은 노드들의 집합이 노드들의 동작을 통합하고 데이터 정보를 전송, 처리하는 것을 허용하는 프로토콜을 수행하기 위해 이러한 기초적인 통신 능력을 사용한다. 센서 네트워크에서 기초적인 무선 전송 능력은 노드들 사이에서 확산되는 데이터 특성과 연관된다. 이것은 하나의 기본 노드가 약간의 구별되는 정보를

가진 패킷을 전송하는 위치에서 유동적인 프로토콜에 의해 수행되어진다. 수신하는 노드들은 패킷을 재전송하고 그 결과 좀 더 멀리 떨어진 노드들은 그것을 수신할 수 있다. 그러나 한 노드는 다수의 주변 노드들로부터 같은 메시지의 다른 버전을 받을 수 있고, 그래서 그 네트워크는 중복된 것을 알아내고 금지시키기 위한 구별되는 제어 정보를 사용한다. 프로토콜이 동작하는 것은 충돌을 피하고 불필요한 전송을 없애는 다양한 기술들을 사용한다.

네트워크는 명령어를 전송하고, 경로를 통해 네트워크 제어를 수행하고 형성하기 위해 분배과정을 사용한다. 또한 라우트를 형성하기 위해서도 분배과정을 사용한다. 각각의 패킷은 기초되는 노드로부터 그것의 거리와 송신기를 구분한다. 분산된 되어진 라우트 트리(route tree)를 형성하기 위해 노드들은 그 루트 근처의 노드 하나를 정의해서 기록한다. 네트워크는 원래의 노드로 거슬러 올라가거나 그 트리 각각의 레벨에서 데이터를 처리함으로써 데이터를 모으기 위하여 이러한 역방향의 통신을 사용할 수 있다.

그 기본 노드는 일부의 고 레벨 작업에 의해 결정되어지는 것처럼 좀 더 강력한 네트워크나 센서네트워크 내의 모이는 접점에서 게이트웨이가 된다. 종종 트리구조와 데이터 수집은 연관되어 있다. 데이터는 부모 노드가 발견되면 그 트리를 따라서 동작을 시작할 수 있다. 노드들은 데이터 메시지를 수신하면서 부모 노드가 될 노드들을 정하게 된다. 네트워크는 최선의 라우트를 강화하기 위해 통계수치를 계속적으로 모은다.

이러한 통신 패턴은 많은 클라이언트 컴퓨터들이 연결을 열어놓고 많은 양의 데이터를 주고받는 인터넷에서 사용되는 방식들과는 다르다. 센서 네트워크에서는 통신은 전체로서 수행되고 참가자들은 물리적 위치나 센서 값들의 범위 같은 특성에 의해 구별되어 정의된다. 라우팅은 노드들이 특성에 의해 데이터에 관심을 표현하는 과정인 유도확산 과정처럼 형성된다. 그 노드들은 라우팅 도표를 형성하기 위해 관심 있는 데이터 종류를 외부방향으로 전송하고, 그것들은 결합되어진 하부구조에서 관심도를 기반으로 기술기 데이터를 연산하고 이것을 증가시키는 방향으로 데이터를 수집한다.

신뢰도는 다양한 패턴에 따라 나타나는데, 센서 네트워크는 신뢰성을 결정하는 데이터 근원지와 전달 목적지 사이의 패킷 매칭이나 바이트를 사용하는 두 지점간의 연결을 설정하는 인터넷과 대조적으로 그것들이 확실하게 흡에서 흡으로 데이터를 전송하는 위치에서 효율적인 네트워크 접근법을 매치해야 한다.

### 3. 메쉬 센서 네트워크 응용 애플리케이션

#### 3.1. 환경 모니터링

삼나무(Redwood)의 상태를 모니터링 한 경우에는 삼나무 전체에 센서 네트워크를 설치하여 세밀한 지점의 기후를 감시하는 것은 전체 나무의 상태를 정확히 파악할 수 있다.

이들의 기후적인 요소는 광합성, 수분과 영양분의 이동, 성장 패턴의 비율을 결정한다. 중요한 변동은 개별적

인 종들의 크기에 관계없이 존재하는 것으로 알려져 있고 연구원들은 극지적인 기후 구조가 나무들의 영역을 넘어 변화하는 것으로 믿고 있다. 또한 그들 자신의 기후 환경을 효율적으로 만드는 나무들의 수분의 운송율과 호흡작용의 규모는 나무 주변의 극지적인 기후에 영향을 줄 것이다. 모든 이러한 요소들은 나무 표면이나 내부에 존재하는 종들의 서식 원동력에 영향을 준다.



그림2. Redwood 센서 네트워크 구성

연구원들은 전통적으로 무게가 30파운드 되는 장비들을 높은 곳에 부착된 크랭크에 의해서 나무위로 올림으로써 삼나무 생태계 측정을 수행해 왔다. 시리얼 케이블은 그때 나무들에 매달려 있고, 지상에 있는 데이터 기록계가 측정값을 모으는 방식으로 테스트 시스템을 구축하였다.

하나의 무선 센서 노드는 얇은 필름용 크기 정도의 튜브 내에 고정되고, 나무의 껍대기에서 두개의 빛을 받아들이는 센서들은 염록소가 민감하게 반응하는 위치에 빛과 광합성에 필요한 요소인 태양 복사를 측정한다. 아래쪽 부분 음지의 개별적인 센서들은 밝은 빛을 측정한다. 밑단에는 상대적인 습도, 기압, 온도를 감시하기 위한 환경적인 센서들이 있다[6][7].

튜브 내에서 날씨로부터 보호 받는 중심부분은 외부환경과 노드들 사이에서 정보를 연결하고 데이터를 처리하고 수집하기 위한 저전력 무선통신 및 소형의 컴퓨터와 배터리를 포함한다. 이 센서 네트워크 노드는 나무의 많은 지점에서 동시에 오랜 시간동안에 걸쳐 데이터를 얻는 저가의 시스템을 제공한다.

그 네트워크의 데이터는 날씨전선이 어떻게 그 나무를 오르내리는지 보여준다. 나무의 정상부분은 태양온도에 비례하여 급격히 증가한다. 이러한 온난기류는 날씨가 따뜻하면 나무 밑으로 내려오고 그것의 정도는 감소해진다. 해질녘에는 반대의 상황이 벌어지고, 정상부분의 온도는 기본 수치 이하로 떨어진다.

습도의 변화는 기후변화보다는 더 중요한 의미를 갖고 있는 데이터라고 한다. 나무 내에서 습도를 증가시키는 매우 많은 양의 물을 이동시키기 때문에 밀집된 장비들의 중요한 예시에서 보듯 이러한 날씨전선은 그러한 대규모에서처럼 영양분을 이동하고 물을 흡수하고 성장이 활발하게 일어나는 것을 이해하는 것에서 관측되어 질수

있는 강력한 온도와 수분의 변화를 만들지만, 여전히 그것들은 몇 개의 간헐적인 장비만으로는 관측되어질 수 없다.

### 3.2. 사물의 움직임 모니터링

사물을 감지하는 것은 다양한 응용 서비스에 사용될 수 있는 가능성이 많다. 대부분 응용 서비스들은 상태를 기반으로 여러 사물, 시설 등을 유지 보수하는 형태처럼 보여 질 수 있다. 기계나 모터, 비행기의 날개 부분 또는 건물들 같은 물리적 구조는 자극반사와 음파의 발산과 진동에서 전형적인 상태를 가지고 있다. 모니터링 중 각 사물이 가지고 있는 일반적인 고유 신호에서 변형이 발생하였다면, 마모 등으로 인한 상태의 변화를 의미한다. 배어링 같은 경우는 마모가 심하게 진행 되었을 때, 사람이 인식할 수 있는 소리나 진동을 발생 시킬 것이다. 센서 네트워크는 이러한 변화를 사람이 느끼지 못할 정도의 세밀한 변화를 무선으로 항상 감지할 수 있다.

좀 더 정밀한 분석을 모니터링을 수행하기 위해서는 센서 노드들의 동작이 시간상 높은 정확도를 필요로 하게 된다. 실제 반도체 설비 공정에 정밀도가 높은 수천 개의 진동 센서 네트워크 노드를 추가하면, 더욱 정확한 공정 상태를 모니터링 할 수 있다. 현재는 설비 관리자들이 센서가 부착된 컴퓨팅 장비를 가지고 공장을 돌아다니며 데이터를 수집, 저장한다. 그런 후 그것들의 마모 신호를 분석하는 중앙컴퓨터에 이러한 데이터들을 옮겨 저장한다. 반면에 무선 센서 네트워크는 보다 편하고 효율적인 방법을 제공한다. 각각의 설비 장치에서 모니터링 데이터를 처리하고 운영하는 부분에 연속적으로 데이터를 전달하는 것이다. 샘플링 비율은 전형적으로 음파분석을 위한 수 킬로와 진동분석을 위한 수 백 Hz의 환경감시에서 하는 것 보다 높다. 이것은 네트워크가 어떻게 샘플링을 수행 하는지에 대해 좀더 자세하고 광범위한 샘플링 특성을 요구한다. 데이터를 임시 저장하는 것은 대용량의 기억장소를 요구하고, 시스템은 취득된 데이터에서 잠재적으로 좀더 확장적인 처리능력을 요구한다.

많은 양의 원본 데이터를 전송하는 것 보다, 센서 네트워크 노드들은 변화된 신호나 진동의 상태를 전달하기 위한 신호 분석을 수행할 수 있다. 센서 노드들은 언제 샘플링 되는지 결정되거나 샘플링 되어질 때 무엇이 활성화 되는지를 모니터링하기 위한 제어 네트워크를 모니터 할 수 있다. 획득되어진 데이터 처리 비용을 제거하고 모니터링 하는 것이 전체의 비용을 줄이기 때문에, 이런 장점은 설비의 운영에서 해당 성능을 향상 시킬 수 있다.

예를 들어 다리, 건물, 항공기 등에서의 구조적인 반응을 분석하는 것은 시공간 분석에서 구조적으로 서로 다른 지점에서 수집된 데이터를 사용하기 위해 더 많은 요구들이 필요하게 된다. 이것은 흔히 노드들 사이에서 매우 정확한 시간 동기화 확립되기를 요구한다. 노드들은 시간과 관계된 원본 데이터나 구조적인 분석을 수행하기 위해 처리된 데이터를 공유한다. 하나의 장비로부터 얻은 센서 데이터는 다수의 다른 지점에서 각각 모델에 근거한 분석을 위한 입력으로 사용되어 질수 있고, 이러한 지점들에서 센서 데이터를 비교할 수 있다. 연구원들은 예외적인

상황을 알아내기 위해 기준이 되는 정상적인 상황에서 그것들을 반복 사용함으로써 이러한 모델들을 재정리 할 수 있다.

### 4. 결론

메쉬 센서 네트워크 기술은, 작고 단순한 하드웨어를 기반으로 소형 소프트웨어 개발을 지향하고 있기 때문에 고급 기술 창조를 통한 신산업의 창출이라는 기술적 기대를 실현하기에는 구현 난이도에 대한 회의적 견해가 많은 것 같다. 그러나 기술적 난이도와는 상관없이 새로운 컴퓨팅 개념의 도출과 실제 구현을 지향하는 기술 개발 방향은 매우 긍정적으로 평가받을 수 있는 추진 방향이다.

메쉬 센서 네트워크 같은 저전력 무선 네트워크가 실제 산업과 생활에 적용되고 상용화되기 위해서는 안정되고 상호운용이 가능한 네트워크 프로토콜의 실제 구현이 요구되는데, 이런 요구사항을 해결할 수 있는 방법은 저그비 규격의 제품이 개발되거나 TinyOS를 기반으로 한 기술구현 등으로 예상된다. IEEE 802.15.4 기반의 하드웨어를 이용한 무선 통신 프로토콜이 TinyOS상에서 구현되었기 때문에 하위 기술에 대한 호환성이 보장되었으며, 지금 부터는 네트워크 부분의 기술 구현이 본격적으로 진행될 것으로 보인다.

메쉬 센서 네트워크의 다양한 산업에의 적용은 관련 산업화를 가속화 할 것으로 기대된다. 단기적으로는 무선 네트워크 시장의 확대를 가져올 것이며, 장기적으로는 유비쿼터스 컴퓨팅에서의 끊임 없는 네트워크를 구성할 수 있는 기반 기술로 사용될 것이다.

### 참고문헌

- [1] <http://www.intel.com/research/exploratory>
- [2] <http://www.chipcon.com>
- [3] <http://www.moteiv.com>
- [4] J. Hill, R. Szweczyk, A. Woo, S. Hollar, D. E. Culler and K. S. J. Pister, System architecture directions for networked sensors, Proceedings of ASPLOS, pages 93-104, Boston, MA, USA, Nov. 2000.
- [5] [http://www.intel.com/research/vert\\_manuf\\_condmaint.htm](http://www.intel.com/research/vert_manuf_condmaint.htm)
- [6] David Culler, Deborah Estrin, Mani Srivastava, Overview of Sensor Networks, IEEE Computer, pages 41-49, Aug 2004.
- [7] [http://today.cs.berkeley.edu/retreat-6-04/buonadonna-NEST\\_06\\_04.ppt](http://today.cs.berkeley.edu/retreat-6-04/buonadonna-NEST_06_04.ppt)